commodore Magazine

AÑO III Núm. 32
Octubre. 1986 300 Ptas.

REVISTA INDEPENDIENTE PARA USUARIOS

Hablemos de ordenadores

Compatibles

Parámetros estadísticos

SUPLEMENTO
PROGRAMAS
12 PAGINAS

Noticias, software, trucos...

GRAFICOS VIC-20·C-64·C-128

:RITEMAN: news

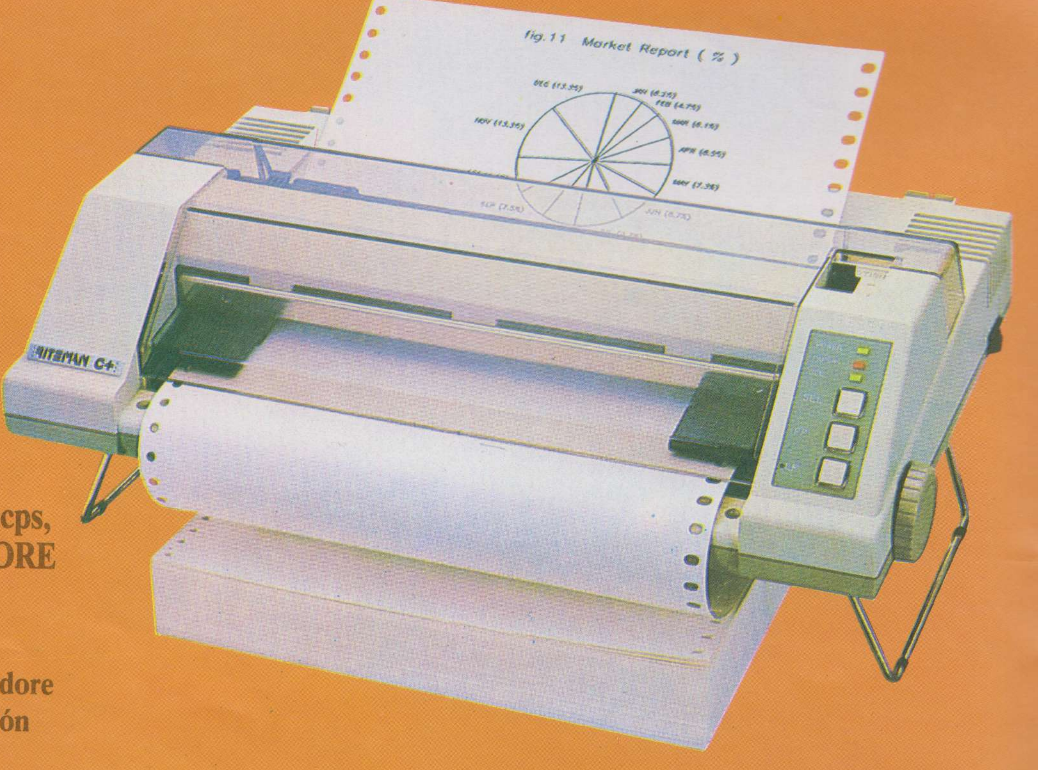
DATAMON REPRESENTACION EN ESPAÑA DE: :RITEMAN:

PROVENZA, 385-387
TEL. (93) 207 24 99*

TELEX 97791
08025 BARCELONA

IMPRESORA PARA SU COMMODORE
(óptima relación precio/prestaciones)

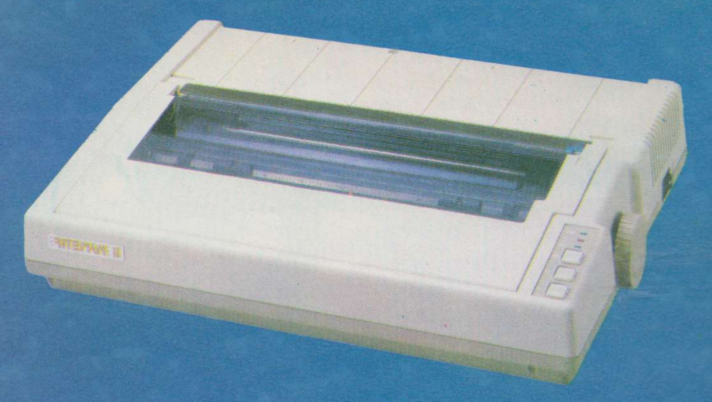
—Cabezal 9 agujas
—Doble operatividad
—Cinta autoretintada
—Tampón retintable
—Ausencia de rodillo
—No dobla el papel
—Elevadores inferiores
—Admite texto rígido
—Máximos tipos de escritura



Modelo SUPER C+, 120 cps, NLQ, ASCII y COMMODORE

• Conexión directa a Commodore (cable incl.) Tracción y fricción

LA IMPRESORA PARA COMMODORE, ASCII Y PC'S COMPATIBLES
(Máxima versatilidad/precio ajustado)



RITEMAN 10-C

—140 cps, tracción y fricción
—Paralelo centronics/Commodore serie DIN
—Tablas ASCII y PC en Rom interna
—Tabla 100% Commodore y 8K RAM en módulo
—Interface Commodore exterior incluido
—RS 232-C opcional

NOTA: Para Aplicaciones en las que se necesite más velocidad, o mayor tamaño de carro, también pueden aplicarse nuestros interfaces externos a los modelos RITEMAN 10/II y RITEMAN 15.

Director:
Rubén Sanz
Redacción:
Teresa Aranda
Colaboradores:
José D. Arias
Alejandro de Mora-Losana
Paloma Saco
Diseño:
Benito Gil
Edita
PUBLINFORMATICA
Bravo Murillo, 377 - 5.º A
Telf.: 733 74 13. Madrid - 28020
Presidente:
Fernando Bolín
Director Editorial
Revistas Usuarios:
Juan Arencibia
Director de ventas:
Antonio González
Jefe de Producción:
Miguel Onieva
Servicio al cliente:
Julia González - Telf.: 733 79 69
Publicidad:
Emilio García
Dirección, Redacción y Publicidad:
Bravo Murillo, 377 - 5.º A
Telf.: 733 74 13
Publicidad Barcelona:
María del Carmen Ríos
Pelayo, 12
Telf.: (93) 301 47 00
ext. 27-28 y (93) 318 02 89
08001 BARCELONA
Depósito Legal: M-6622-1984
Distribuye: S.G.E.L.
Avda. Valdelaparra, s/n.
Alcobendas. Madrid
Distribuidor en Venezuela:
SIPAM, S. A.
Avda. República Dominicana
Edif. FELTREE
Boleita Sur Caracas (Venezuela)
Distribuidor en Argentina:
DISA
Sta. Magdalena, n.º 541
Buenos Aires (Argentina)
Fotocomposición: Consulgraf
Nicolás Morales, 34. 28019 Madrid
Fotomecánica: Karmat
Pantoja, 10. Madrid
Imprime: G. Velasco, S. A.
Solicitado control O.J.D.
Esta publicación es miembro
de la Asociación de Revistas
de la Información , asociada
a la Federación Internacional
de Prensa Periódica FIPP.
El P.V.P. para Ceuta, Melilla y
Canarias, incluido servicio
aéreo, es de 300 ptas. sin IVA

Rogamos dirijan toda la correspondencia relacionada con suscripciones a Commodore Magazine. Edisa. Tel.: 415 97 12. López de Hoyos, 141-5. 28002 Madrid. Para todos los pagos reseñar solamente Commodore Magazine.
Para la compra de ejemplares atrasados dirigirse a la propia editorial Commodore Magazine, Bravo Murillo, 377 5º A. Tel.: 733 74 13. 28020 Madrid.

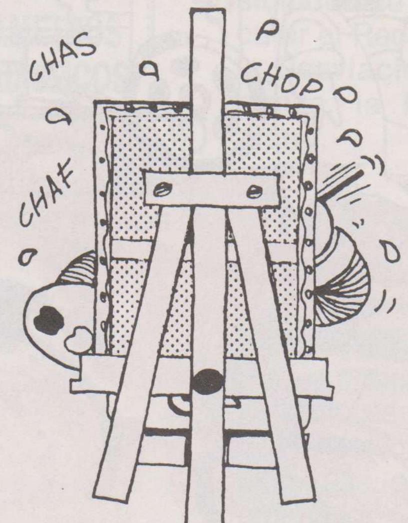
Editorial

Los gráficos diseñados por ordenador marcan una nueva era en el arte de la imagen. El cine y el vídeo se informatizan utilizando imágenes diseñadas digitalmente para la realización de los efectos especiales en los temas fantásticos y de ciencia-ficción.

Los nuevos ordenadores poseen cada vez una mayor definición gráfica, el número de pixels en pantalla aumenta espectacularmente al igual que la paleta de colores gráficos, permitiéndonos generar imágenes de alta calidad.

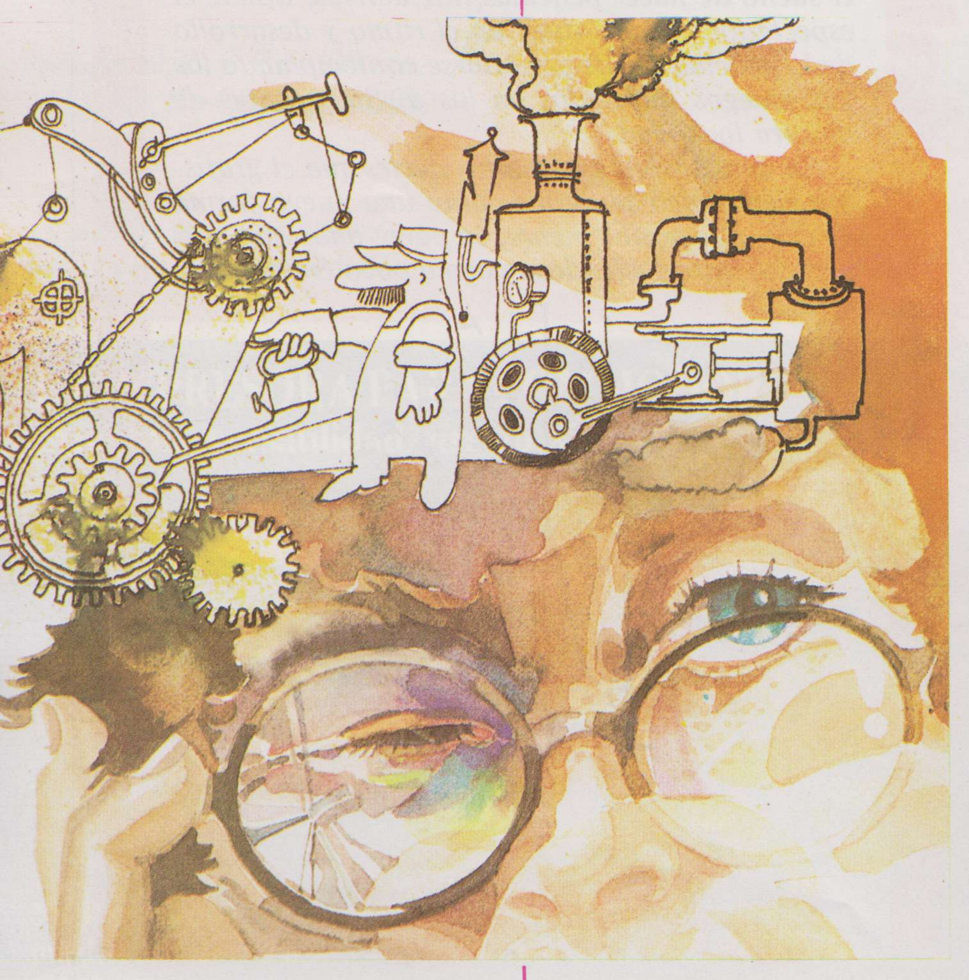
La animación gráfica comienza a ser realidad y el sueño de hacer películas interactivas, donde el espectador pueda influir en el ritmo y desarrollo de la película puede ya intuirse contemplando los videojuegos instalados en los distintos bares de nuestra localidad.

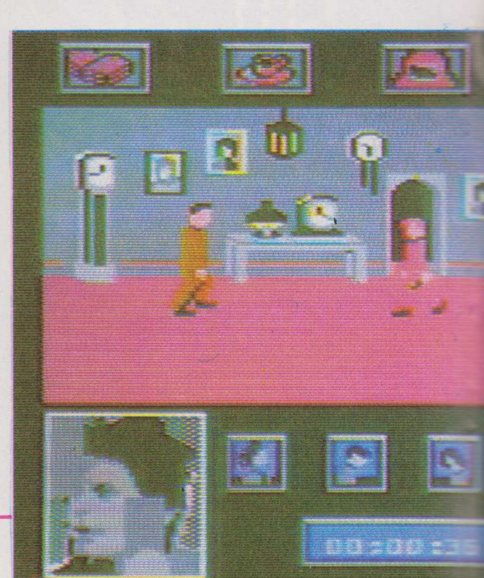
En conclusión, podríamos decir, que el grafismo electrónico se define como una nueva forma de arte, que posiblemente, en pocos años, nos sorprenda, introduciéndose poco a poco en todas las artes.



Esta revista no mantiene relación de dependencia de ningún tipo con respecto de los fabricantes de ordenadores Commodore Business Machines ni de sus representantes.

Sumar





rio





NOTICIAS

Personal computer World Show

En esta importante feria londinense, celebrada durante los primeros días del mes de septiembre, se han presentado las dos nuevas estrellas de la firma INDESCOMP.





La «Máquina Alucinante».
NUEVO SPECTRUM 128K + 2.

La «máquina alucinante» como se ha dado en llamar a la última creación de la firma SINCLAIR-AMSTRAD, tendrá un precio aproximado de 33.000 ptas. más I.V.A., según fuentes de INDESCOMP consultadas durante la celebración en Barcelona del SONIMAG, donde hemos podido contemplar el nuevo equipo «in situ».

Entre sus configuraciones técnicas más interesantes, destaca un RAM de 128K y ROM 32K., en la CPU se mantiene el conocido Z-80A, una resolución de 256*192 pixel, 24*32 resolución de colores, 8 colores para texto y 8 para papel, border controlable independientemente, teclado QWERTY de 58 teclas, cassette integrado, tres voces musicales con o sin ruidos con 16 envolventes, salida de sonido a TV o audio, port de UHF, port impresora serie, monitor RGB, Keypad, MIDI, dos joysticks (de configuración exclusiva), bus de expansión de 10 para el Z80, software compatible con el 48K y 128K, reset incluido y transformador de 9 voltios de potencia.

AMSTRAD PC 1512,
LA REVOLUCION DEL MERCADO.

El nuevo PC de Amstrad ha sido presentado con todo lujo de detalles, y a priori parece poseer todos los requisitos necesarios para lograr un impacto importante en el mercado informático.

Si dejamos a un lado su precio (entre 399 y 799 libras), elemento que por sí sólo hablaría de brillantes expectativas en el mercado, y nos centramos en el equipo, encontraremos en su interior un procesador 8086 de 16 bits, 512 Kbytes de memoria ampliables a 640 K., reloj de cuarzo en tiempo real con batería de seguridad, interface RS232c y paralelo, 85 teclas, ports de joysticks, ratón, discos de 5 1/4 pulgadas con 360K., monitor monocromo y color, 80 columnas y 16 colores.

Software compatible con Microsort MS DOS 3.2 y Digital Researchs DOS Plus, GEM Desktop, GEM Paint y Basic 2, completan la tabla de compatibilidades de un aparato basado en aprovechar el soft existente en el mercado profesional.

Novedades

DIGIDRUM.

El Digidrum es un periférico que, conectado al Commodore 64 o 128, permite transformar en una potente batería electrónica programable de prestaciones y características similares a las profesionales.

Tiene capacidad para memorizar un total de 50 patrones distintos y el correspondiente ensamblado de los mismos para formar un total de hasta 10 canciones.

Hay que remarcar muy especialmente que el Digidrum recrea los sonidos mediante muestras digitalizadas y prescindiendo del generador interno de sonido del Commodore, por lo que la calidad resultante es realmente excepcional. Hay una librería disponible de más de 100 sonidos distintos y que pueden cargarse en bloques de 8 o individualmente. Se suministra con manual de instrucciones, un diskette con el programa para hacerlo funcionar y una selección de sonidos y el cartucho enchufable en el Commodore en el que se localiza la salida de audio y una de disparo externo.

MICROVOX.

Ahora es posible digitalizar el chirrido de una puerta, el motor de un coche, el ladrido de un perro o la voz de una persona diciendo «hola» o haciendo gárgaras, y tocar el sonido resultante como si de un instrumento musical se tratara.

Utilizando el editor de forma de onda, se podrá modificar la muestra y reproducirla en sentido normal o invertido, transportándola a cualquier tono en un rango de cuatro octavas (vía teclado MIDI o de ordenador). Se puede almacenar hasta 16 voces en la memoria y acceder a ellas a través del secuenciador de 2.000 notas que se incluye en el software de la unidad. También pueden almacenarse las muestras preferidas en diskettes.

El Microvox permite generar otros efectos especiales como Armonización, Flanger/Phasing, Eco y la toma de muestras múltiples o extremadamente largas de hasta 19,9 segundos.

Novedades ABC SOFT

SLAM BALL.

Un Pimball que te permite dar efecto a la bola y con un excelente scroll de pantalla que simula el tamaño real de una máquina. Intenta realizar muchos puntos apagando las dianas y pasando por los pasillos y conseguirás apuntar tu nombre en la tabla de records.

NEW YORK CITY.

Recorre la ciudad de New York, visita Central Park, The World Trade Centre o el City Bank, ten cuidado y no te des un golpe con el coche, pues tendrás que ir andando para solucionar esta divertida historia.

BREAKDANCE.

Simulación del característico baile «Breakdance», donde tú puedes seleccionar entre cinco juegos distintos el tipo de competición que deseas realizar, acompañado en todo momento de una buena música «break».

Amiga compatible

Commodore ha anunciado un accesorio para su modelo Amiga que permite ejecutar aplicaciones estándar PC-DOS.

Este subsistema, que lleva el gráfico nombre de «Sidecar Amiga», está diseñado para introducir al Amiga en entornos de negocios, donde la arquitectura IBM-Intel es un requirimiento imprescindible.

El Sidecar consiste en un microprocesador de Intel, el 8088, con una velocidad de reloj interno de 4,77 Mhz, un disquete de d5 1/4'', un suministrador de potencia independiente, tres slots de expansión para añadir tarjetas del PC y 256 o 512 Kb de memoria. El entorno DOS creado por el Sidecar ejecuta tareas bajo el sistema operativo multitarea del Amiga, conocido como Amiga-DOS. Cuando el usuario vuelve al Amiga, la máquina se sitúa en un menú preguntando al usuario si se requiere una sesión del DOS. Si es así el usuario solicita la sesión del DOS y puede comenzar a cargar programas desde la unidad de disquetes de 5 1/4'' del Sidecar.

Según Gail Wellington, especialista técnico de Commodore, los usuarios que ejecuten simultáneamente aplicaciones de PC-DOS y del Amiga no sufrirán en sus prestaciones, porque tanto el Sidecar como el Amiga tienen sus propios procesadores y memoria.

Los usuarios pueden transferir datos sobre las sesiones del AmigaDOS y del PC-DOS del Sidecar. Asimismo, este accesorio del Amiga puede llegar hasta 2 Mb de memoria, instalar un disco duro de 20 Mb y un coprocesador matemático opcional, el Intel 8087.

Gráficos en el Vic-20, C-64 y C-128

Los podemos apreciar en los vídeojuegos. Los intentamos crear introduciéndonos en el mundo mágico de los sprites. Los gráficos nos presentan la parte más atractiva de la programación, y la que deparará un mayor uso en aplicaciones profesionales y artísticas.



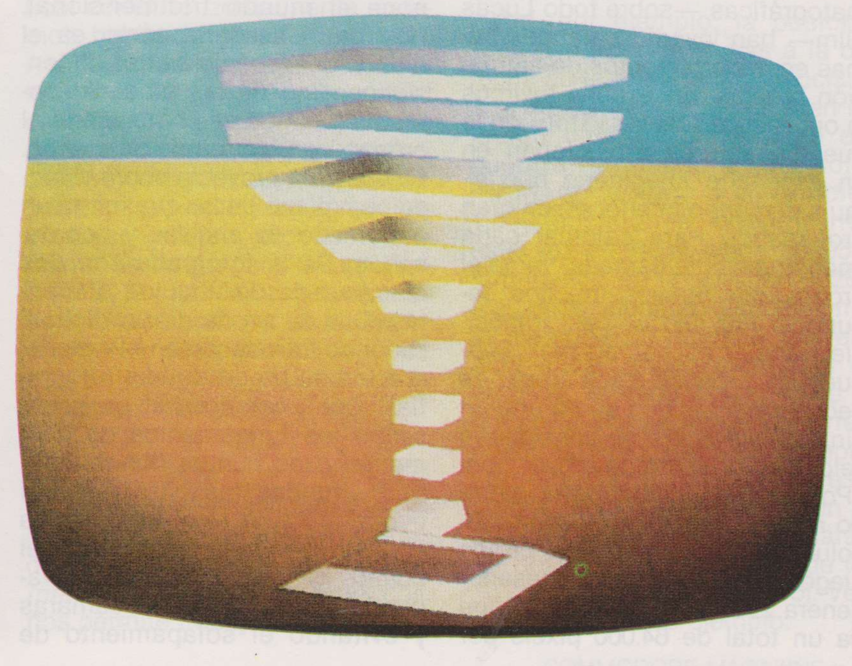
¿Cómo se genera esa imagen, ese gráfico que visualizamos en la pantalla del monitor o de la televisión? En el caso de los monitores monocromáticos, en el interior del tubo de rayos catódicos se encuentra un «cañón electrónico», basado en un filamento calentado por la corriente que lo atraviesa, que produce un haz de electrones. Este haz guiado por los campos eléctricos y magnéticos se desplaza de derecha a izquierda y de arriba a abajo, aplicando mayor o menor intensidad a cada uno de los puntos de la pantalla que ha sido recubierto por una fina capa de fósforo provocando una mayor o menor luminosidad. La imagen se reconstruye punto por punto a una velocidad de barrido de pantalla de un cincuentavo de segundo. En esa mínima fracción de tiempo es posible recorrer toda la pantalla.

Las pantallas de color basan su funcionamiento en un fenómeno óptico: todos los colores se pueden generar a partir de tres denominados fundamentales. Estos son el rojo, el verde y el azul. En el interior del tubo existen tres «cañones electrónicos» en lugar de uno. La superficie del tubo permanece recubierta de miles de puntos de fósforo dispuestos en tres grupos, siendo cada uno de ellos capaz de emitir luz roja, verde o azul. Con idéntico movimiento descrito anteriormente se desplazan ahora tres haces en lugar de uno y mediante una técnica denominada síntesis aditiva (la combinación de los tres colores básicos) proporcionan una sola imagen en colores.

Es así como el ojo humano puede percibir las distintas tramas de colores. Los gráficos se han hecho tan habituales en la programación de juegos, histogramas, representación de figuras geométricas, etc., que prácticamente los avances que se van consiguiendo en este terreno pasan desapercibidos.

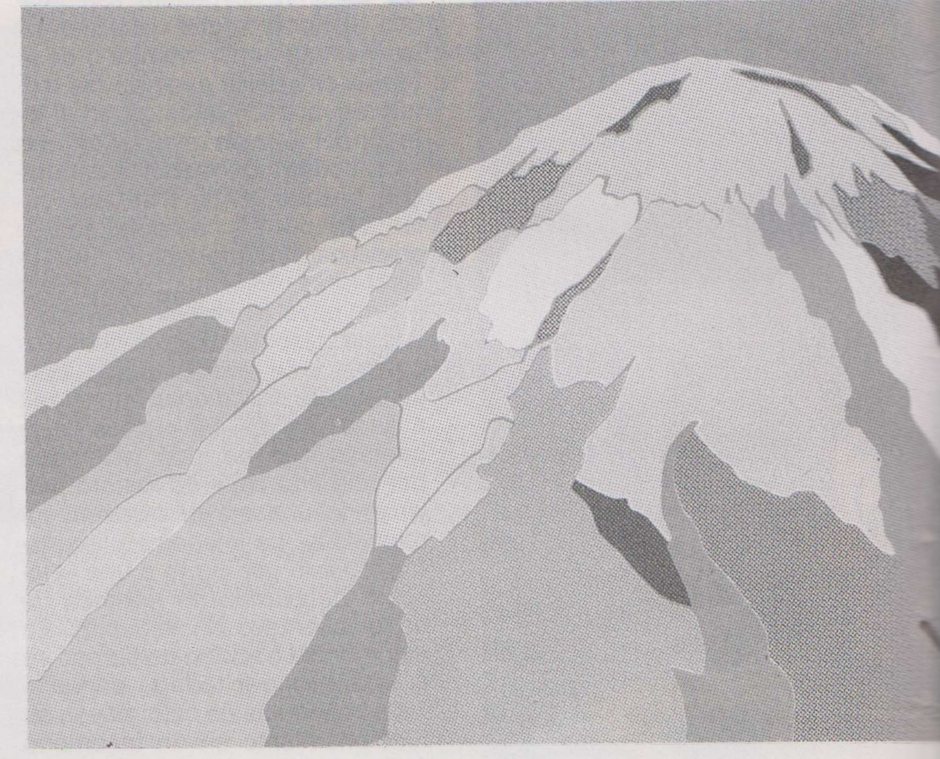
Una comprobación sencilla. Recordad los gráficos de algunos juegos de hace dos años. Comparadlos con los actuales. ¿Se aprecia la diferencia?

Pero si aumentamos la complejidad del ordenador, si aumentamos su capacidad, no debemos hablar ya de gráficos, sino (más propiamente) de procesamiento digital de las imágenes. En un primer momento las áreas de este



procesamiento eran más bien restringidas; intentar clarificar la información gráfica para que pudiera ser interpretada por el hombre y procesar las imágenes reales para que fueran percibidas automáticamente por las máquinas. Actualmente el campo de aplicaciones crece sin cesar. Imágenes procedentes de sondas espaciales se procesan con el propósito de corregir algunas distorsiones. Es posible con esta técnica intensificar y decodificar toda la información procedente de la investigación espacial. Los médicos y los geógrafos pueden contrastar, codificar e interpretar las imágenes obtenidas mediante rayos X. Aplicaciones similares se pueden encontrar en Astronomía, Biología, Medicina oncológica, aplicaciones industriales, etc.

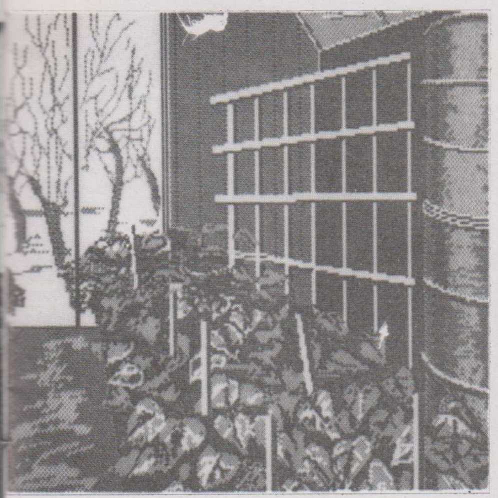
Las imágenes también pueden tener un componente lúdico y artístico. El nuevo arte se llama C.G.I. *(Computer Generated Imagery)*. Imagery se puede traducir por conjunto de imágenes, pero también es imaginación, fantasía. Fantasía que toma forma de presentación de programas o de anuncio publicitario creado por ordenador. Las productoras cinematográficas —sobre todo Lucas Film— han invertido grandes sumas en departamentos de animación propios. En España tuvimos la oportunidad de ver «Tron» en la que los actores evolucionan en un escenario imaginario que simula el mundo interior de un gran procesador. Para calcular cada imagen aislada, llamada *frame* el procesador precisa muchos segundos. Por ello es casi imposible trabajar en tiempo real. Después de producir cada toma, es necesario grabarlos uno a uno en cinta de vídeo. Así se consigue la velocidad correcta de proyección. ¿Por qué no generarlos en tiempo real? El problema es claro: resolución de las imágenes. En los juegos comerciales el procesador genera 200 × 320 puntos, lo que da un total de 64.000 pixels por



cada imagen. Para intentar reproducir la calidad cinematográfica el ordenador tiene que manejar 4.000 × 6.000 u 8.000 × 8.000 pixels. La friolera de 64 millones puntos. Otro problema es el de la perspectiva. El ordenador no conoce el mundo tridimensional. Hay que enseñarle cómo es el mundo del espacio en tres dimensiones. Uno de los procedimientos empleados es comunicarle al ordenador la forma de un cuerpo. Para ello se proyecta sobre el modelo un plano de luz que forma un entramado de abcisas y coordenadas. Se le fotografía con dos cámaras desde ángulos diferentes. Con la ayuda de un digitalizador se introducen en el ordenador los datos de ambas fotografías. Con posterioridad se le enseñan los fundamentos de geometría y los puntos donde estaban situadas las cámaras. Así puede efectuar el cálculo de las coordenadas tridimensionales del objeto. El proceso se repite, variando el ángulo de las cámaras y evitando el solapamiento de imágenes. Si para las mediciones se utiliza un rayo láser, además de los valores de luminosidad del objeto, se obtienen datos sobre la distancia a la que se encuentra cada punto. Supongamos que deseamos que la imagen se mueva. Para que las órdenes de animación se cumplan es preciso introducir los datos sobre rotación, traslación y ampliación (ejes x,y z). Con esa nueva información el ordenador calcula una matriz de 4 × 4. Una vez obtenida la matriz, la aplica a todos los puntos angulares del modelo y encuentra la nueva posición de los puntos. Las operaciones que debe realizar para cada pixel son veinticuatro. Ahora hay que informar a la máquina sobre las imágenes-clave y donde queramos que aparezcan. El ordenador genera las imágenes intermedias.

Tenemos ya nuestra imagen tridimensional, pero el ordenador ha de formar una imagen de dos dimensiones para el monitor. La parte fácil es determinar la perspectiva: mayor o menor profundi-





dad de campo. Pero también debe calcular los colores y reflejos de los objetos no ocultos, determinar las superficies ocultas, representar figuras transparentes o reflectantes, etc. Observando la complejidad de las operaciones nos podemos hacer una idea de la capacidad de cálculo que precisa un ordenador para crear una imagen animada. Este nuevo arte (o ciencia-arte) prestará al cine, la fotografía, el vídeo, y en general a todas las artes como componente visual, unas posibilidades insospechadas.

MODOS GRAFICOS

Pero descendemos al mundo de los micros. Existen diferentes formas de visualizar los gráficos del ordenador. Hablamos de modo texto o página texto para referirnos a aquel modo que podemos contemplar al encender el ordenador. La configuración habitual Commodore (si exceptuamos el VIC) es la de 25 líneas por 40 columnas. Una característica peculiar del modo texto es que los distintos puntos de la imagen no pueden iluminarse ni ser apagados independientemente, sino sólo en base a esquemas prefijados, es decir, según el juego de caracteres. Cuando el ordenador opera en modo gráfico cualquier punto de la pantalla puede ser iluminado o apagado individualmente. Podemos así, visualizar en pantalla cualquier dibujo que nos interese, iluminando (de forma correcta) los puntos precisos.

Si nos referimos al número de puntos en que está dividido el espacio del modo gráfico, tanto vertical como horizontalmente, estamos hablando de la resolución. Así podemos hablar de una resolución de 320 × 200, como en el caso del C-64 o del C-128. La base de todo lo que puede representarse en modo gráfico o página gráfica es el *pixel* (contracción de las palabras «picture element»; elemento de la imagen). Si en información la cantidad más pequeña es el bit, en gráficos no puede visualizarse nada inferior al *pixel*, que se identifica mediante coordenadas cartesianas.

C-64

Todas las posibilidades gráficas del Commodore provienen del chip denominado VIC-II (*Video Interface Chip* 6567). Este chip nos permite trabajar en modo texto en un formato de 40 columnas por 25 líneas, en alta resolución la presentación en pantalla es de 320 × 200 puntos (64.000 pixels). Al hablar de gráficos es necesario mencionar las memorias de pantalla y de color.

El contenido que nos presenta el C-64 en la pantalla está guardado en la memoria interna del ordenador. La pantalla del mismo es capaz de contener 1.000 caracteres. Habrá por lo tanto 1.000 posiciones de memoria sucesivas. Cada una de estas posiciones tiene 8 bits. Es decir, habrá 256 posibles valores para cada posición de memoria (cualquier número comprendido entre 0 y 255). Esto es lo que se denomina memoria de pantalla. Realizando un **POKE** en la posición adecuada y con el valor de un carácter determinado, lograremos representar ese carácter en la posición que nosotros hayamos POKEado. La memoria de pantalla en el C-64 comienza en la posición de memoria 1024 y acaba en la posición 2023. La posición 1024 es la esquina superior izquierda de la pantalla. La posición 1025 es la siguiente posición situada a la derecha de la anterior y así sucesivamente.

Así, por ejemplo, la posición 1063 es la posición más a la derecha de la primera fila. Asociada a la memoria de pantalla y en relación directa con esta se encuentra la memoria de color (RAM de color). Son también 1.000 posiciones, comenzando esta vez en la posición 55296 y terminando en la posición 56295, siguiendo idéntica disposición a la memoria de pantalla. La relación entre ambas memorias es unívoca. A cada posición de la memoria de pantalla le corresponde un único valor de memoria de color. Cada posición de esta última tiene cuatro bits, con lo que podemos visualizar el carácter deseado en cualquiera de los 16 colores disponibles (0-15). Veamos un pequeño ejemplo:

```
10 PRINT «  »
20 POKE 1514,81
30 POKE 55786,7
```

Esto es: 1514 es una de las posiciones de memoria, 81 es el código (ver juego de caracteres en el manual) de un círculo relleno, 55786 es la memoria de color para 1514 y el 7 es el código del color amarillo. Experimentando y jugando adecuadamente con estas posiciones de memoria podemos lograr efectos curiosos y sorprendentes. Para ello comienza por dibujar, o fotocopiar, un pequeño mapa de la pantalla en el que situarás las posiciones de las memorias de pantalla y de color. Evita las sumas innecesarias de la siguiente forma:

```
POKE 1024 + C
POKE 55296 + C
```

Es decir, sumando a los valores iniciales, o aquellos valores que hayas situado en el margen izquierdo de tu mapa, el valor de la columna (0-39). La suma la puedes dejar indicada en programa.

Algunos ejemplos (mínimos):

```
10 PRINT
20 FOR I = 0 TO 20
30 POKE 1434 + I,99
40 POKE 55706 + I,8
50 NEXT I
```

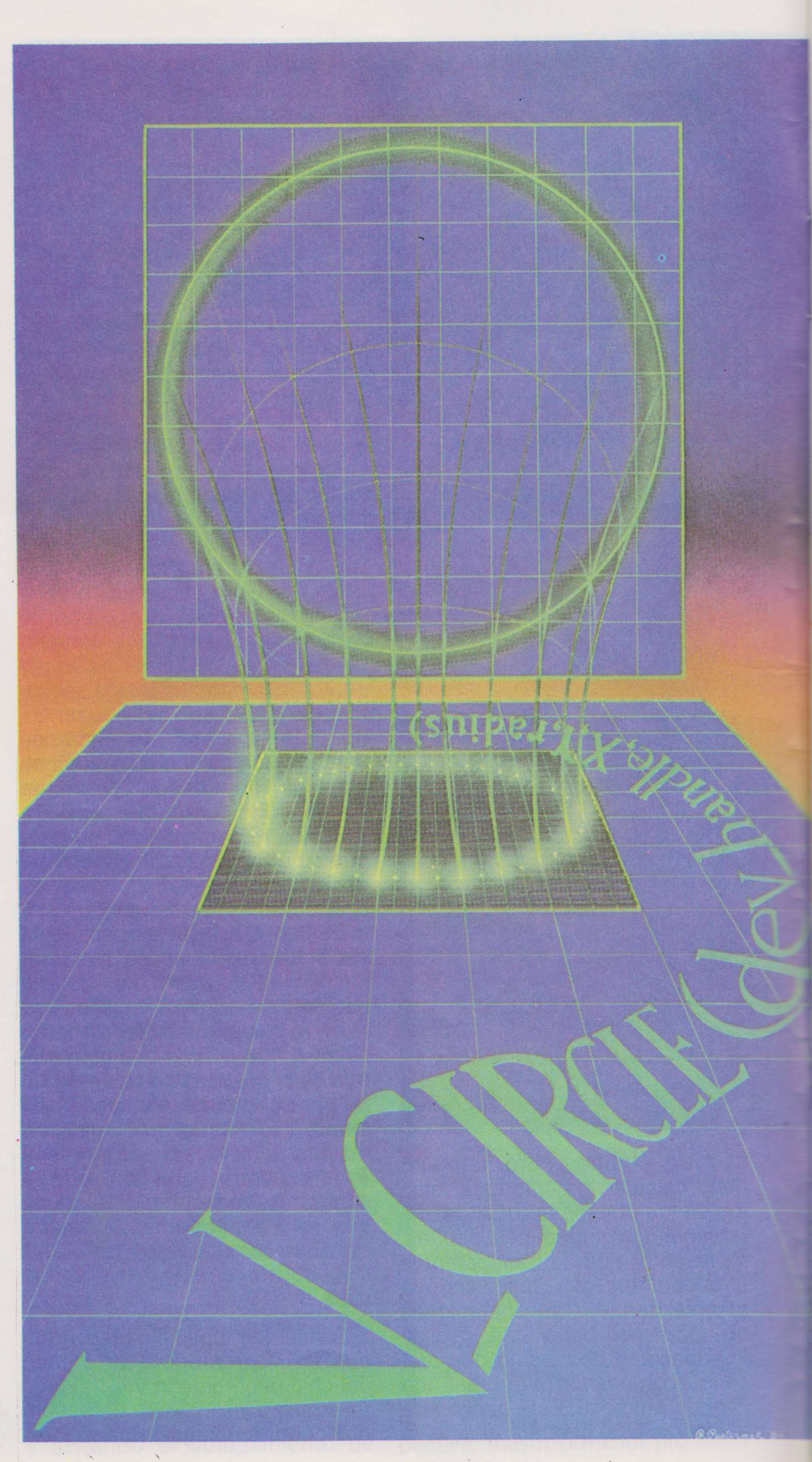
Procura variar el código (99), introduce algún bucle de retardo e intenta variar la posición de inicio del dibujo. Añade las siguientes líneas al programa anterior:

```
60 FOR I = 0 TO 10
70 POKE 1473 + I*40,101
80 POKE 55745 + I*40,8
90 NEXT I
```

Intenta completar la esquina del rectángulo situando correctamente tanto la posición inicial como el código.

Intenta mejorar el aspecto de esta diagonal:

```
10 PRINT «  »
20 FOR I = 0 TO 10
30 POKE 1284 + 41*I,77
40 POKE 1284 + 54272 + 41*I,1
50 NEXT I
```



Por último, un pequeño programa que puedes utilizar como presentación a algunos de tus juegos:

```
10 PRINT «  »
20 POKE 53281,0:POKE 53280,0
30 FOR I = 1 TO 300
40 C = INT (RND (0*15) + 1
50 L = INT (RND (0*1.000) +
   1024
60 POKE L,46
70 POKE L + 54272,C
75 FOR R = 0 TO 200:NEXT R
80 NEXT I
```

Recordar que la posición de la memoria de la pantalla la podemos modificar mediante un **POKE 53272** (registro de control de memoria del VIC-II). Los cuatro bits superiores controlan la dirección de base de la matriz de vídeo. La memoria de color no se puede cambiar. Siempre se haya situada en las posiciones **55296-56295** ($D800-$DBE7).

DEFINICION DE CARACTERES

Sucede, a veces, que deseamos modificar uno o más caracteres de los que habitualmente disponemos en el ordenador. Para ello precisamos realizar lo que se denomina definición de caracteres y que nos permite crear un conjunto de símbolos que han de corresponderse con las teclas existentes en el teclado. La forma de los caracteres (construido por una serie de 8 bytes para cada carácter) se encuentran guardados en la memoria ROM. Puesto que la memoria ROM no se puede modificar, parece que nuestro propósito inicial se presenta como imposible. Sin embargo, la posición de memoria que indica el VIC-II donde buscar los caracteres es un registro que podemos programar. Modificando el puntero de la memoria de caracteres, podemos programar el nuevo juego. Hemos de transferir las imágenes de los antiguos caracteres a una zona RAM. Comencemos el proceso. Lo primero que hay que realizar es dibujar una cuadrícula de 64 cuadrados. Cada carácter está formado por una parrilla de 8 × 8 cuadrados o puntos mínimos. Supongamos que deseamos disponer en nuestro teclado de la letra «ñ». En nuestra cuadrícula de 8 × 8 vamos completando nuestro carácter o símbolo deseado. Utilizaremos el 0 para los puntos apagados y el 1 para los encendidos. Tendremos 8 líneas (bytes). El primer byte (en nuestro caso) contiene el número binario 00111100, es decir, el número 60 en notación decimal. Efectuando las correspondientes transformaciones obtendríamos una serie de 8 números que serán: 60, 0, 194, 162, 146, 138, 134, 0.

Nuestro programa quedaría así:

```
5 PRINT CHR$ (142)
10 POKE 56,48 POKE 52,48
```

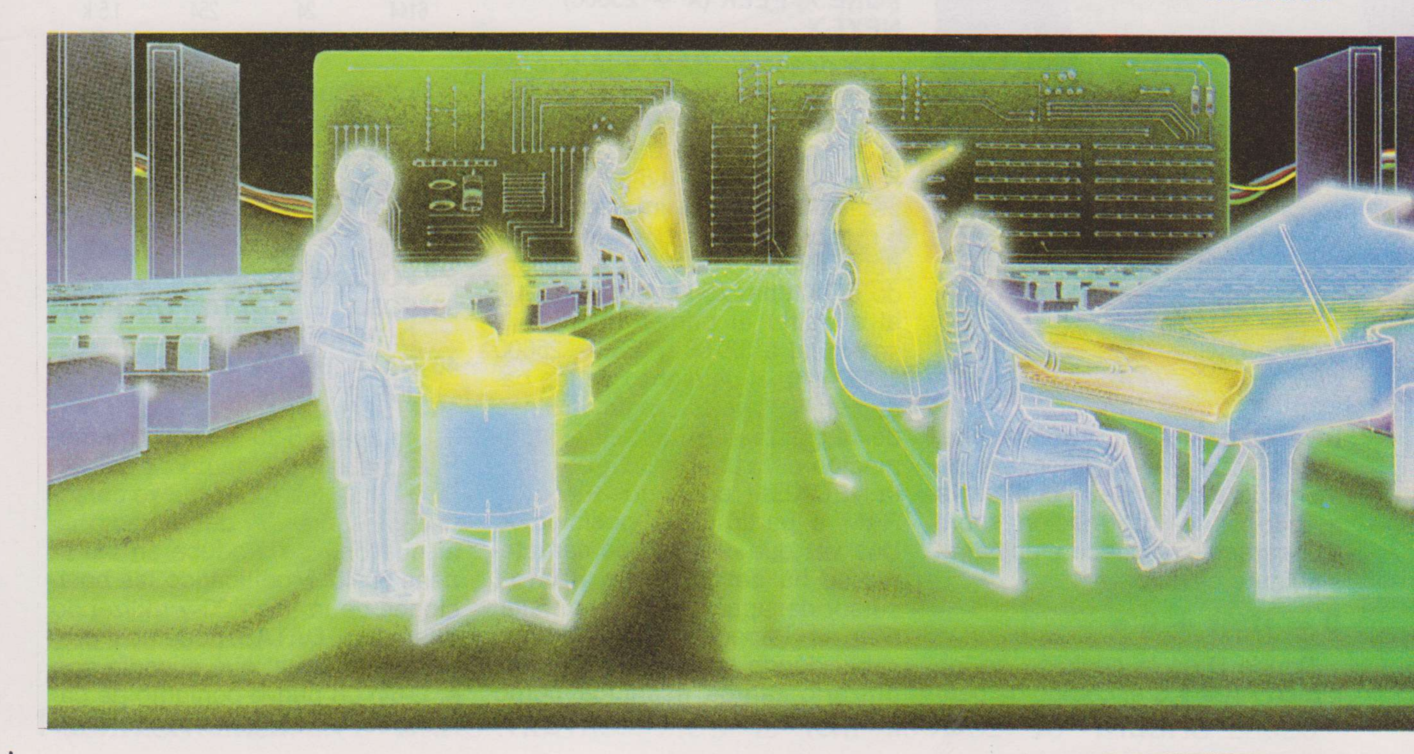
Selecciona el juego de mayúsculas y sitúa el final de la memoria reservada en la dirección 12288 ($3000).

```
20 POKE 56334,PEEK (56334)
   AND 264
```

Deshabilita la exploración del teclado.

```
30 POKE 1, PEEK (1) AND 251
```

Excluye los registros E/S para habilitar la ROM de caracteres.



Gráficos



```
40 FOR I = 0 TO 511:POKE I +
   12288,PEEK (I + 53248):NEXT
```

Transfiere los primeros 64 caracteres de la ROM a la RAM a partir de la dirección 12288.

```
50 POKE 1, PEEK (1) OR 4
```

Excluye la ROM, rehabilitando la E/S.

```
60 POKE 56334,PEEK (56334)
   OR 1
```

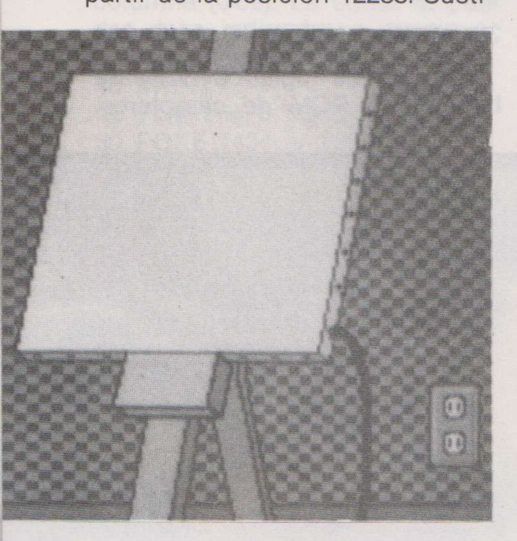
Rehabilita la exploración del teclado.

```
70 POKE 53272 (PEEK (53272)
   AND 240) + 12
```

Sitúa el principio del mapa de caracteres en 12288.

```
80 FOR I = 12288 TO 12288 +
   7:READ A: POKE I,A:NEXT I
90 DATA 60, 0, 194, 162, 146, 138,
   134, 0
```

Lee los bytes que forman el nuevo carácter y los escribe a partir de la posición 12288. Sustituye el carácter @ por la eñe.



Para calcular la dirección de memoria a partir de la cual se va a recoger la información sobre ese carácter se puede emplear el siguiente cálculo:

D = 12288 + C*8

Donde **D** es la posición de inicio en RAM y **C** el código de pantalla. En nuestro caso el código de pantalla de @ era **O**, para otros caracteres consultar el Manual. La línea 80 quedaría, en sentido más amplio, como **FOR I = 12288 + C*8.** Siguiendo un procedimiento similar podemos no sólo modificar unos caracteres, sino incluso todo el teclado. Podríamos obtener símbolos griegos, símbolos químicos, caracteres cirílicos, o caracteres castellanos (¿, ñ, ´, etc.).

VIC-20

El Vic, al igual que sus hermanos mayores, nos brinda la posibilidad de definir nuestro propio juego de caracteres. Existen dos posiciones en memoria RAM que actúan como punteros e indican al ordenador dónde ha de obtener la información del carácter. Podemos alterar esas dos posiciones hacia un área RAM (en el VIC sin ampliar), y situar el nuevo juego de caracteres ahí. Necesitamos pocas líneas para programar parte de la memoria RAM con la información procedente de la ROM:

```
FOR X = 7168 TO 7679
POKE X, PEEK (X + 25600)
NEXT X
```

La dirección RAM a la que se transfiere la información de los caracteres es la representada por la **X.** En este caso utilizamos 0,5 K de la RAM (De 7168 a 7679). Podemos acceder a un área mayor o menor empleando valores superiores o inferiores de **X** y cambiando el número 25600 por otro, que al añadirlo al primer valor de X sea igual a 32768 (7168 + 25600 = 32768), el comienzo de los caracteres ROM. La segunda línea del programa «PEEKea» la información del carácter de la ROM y la «POKEs» en la localización RAM a partir de la posición **X.** Una vez que el juego de caracteres se ha desplazado a RAM el cursor desaparece de la pantalla. Ello no impedirá que podamos saber dónde se encuentra y desplazarlo tantos espacios como precisemos. Hemos de escoger el emplazamiento RAM con mucho cuidado. No hemos de utilizar ninguna ampliación de memoria con este propósito, debido a que los punteros del chip del VIC no tienen acceso a este área. Otras dos áreas que no debemos emplear para almacenar el nuevo juego de caracteres son O (utilizada por el sistema operativo) y la 4096 (comienzo de los programas BASIC).

La mejor localización para los caracteres es el área 4096-7679, donde se almacenan los programas BASIC.

Direcciones de la RAM donde podemos emplazar nuestros nuevos caracteres:

Posición inicial	Poke 52 y 56 con valores	POKE 36969 con los valores	Memoria para el nuevo juego
5120	20	253	2.5 k
6144	24	254	1.5 k
7168	28	255	.5k

Supongamos que queremos disponer del símbolo Ω. El programa quedaría:

```
 5 POKE 52,28:POKE 56,28 CLR
10 FOR X = 7168 TO 7679:POKE
   X,PEEK (X + 25600):NEXT X
20 POKE 36869,255
30 FOR L = 7168 TO 7175: READ
   D: POKE L,D: NEXT L
40 DATA 60, 66, 66, 66, 66, 36,
   195, 0
```

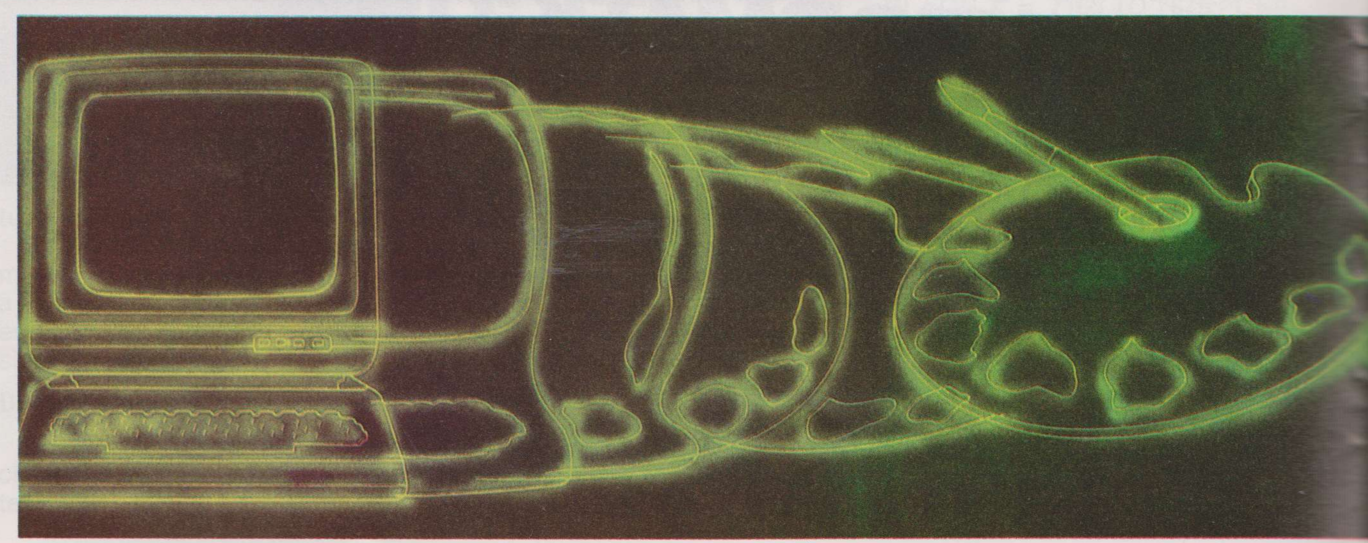
Las líneas 5-20 reservan memoria, cambian los punteros del carácter y desplazan el nuevo carácter. La alteración del símbolo @ al nuevo Ω se produce en las líneas 30-40. Modificando las posiciones de la línea 30 y utilizando diferentes **DATAS** podemos programar (al igual que sucedía con el C-64) un nuevo conjunto de caracteres. Unicamente conociendo

GUIA PRACTICA

la posición inicial y final de la RAM (ver cuadro) y conociendo los Datas, así como el código de pantalla del carácter.

ALTA RESOLUCION

La pantalla del VIC visualiza 22 caracteres por 23. Esto significa que la pantalla tiene una resolución de 176 × 184 pixels, o lo que es lo mismo 32.384 pixels. Ya sabemos que el área mínima que podemos controlar es un pixel que se corresponde con 1/64 del tamaño de un carácter. Ello implica que para controlar toda la pantalla del VIC precisamos de 4048 bytes. En un VIC sin ampliar esto es prácticamente imposible. Lo que podemos hacer, no obstante, es hacer un bit-map de parte de la pantalla. Un área de 8 × 8 caracteres (64 × 64 pixels) requiere únicamente 512 bytes de memoria para almacenar esta información. Supongamos que deseamos combinar caracteres creados por nosotros y gráficos en alta resolución. El primer paso sería proceder a limpiar la parte de la memoria que utilicemos para el «bit-mapping». La memoria de pantalla se haya localizada encima del área de almacenamiento de programas BASIC, desde la posición 7680 hasta la 8191. Para limpiar este área de los bits aleatorios que se almacenan allí, utilizaremos:



FOR S = 7168 TO 7679:POKE S,O: NEXT S

Esto sitúa los bits a cero en la memoria de pantalla comprendida entre 7168 hasta 7679.

Un gráfico en alta resolución precisa de fórmulas para situar y controlar los pixels individualmente. La siguiente fórmula calcula el código de representación del carácter en el interior de la retícula de 8 × 8:

C = INT (X/8)*8 + INT (Y/8)

La **C** representa el código de representación del carácter, mientras que X e Y son las coordenadas horizontal y vertical del pixel a modificar.

La fila del pixel en el interior del carácter se calcula mediante la fórmula:

R = (y/8 - INT (y/8)) * 8

La variable **R** es el número de fila, y la **y** la coordenada vertical. Para calcular el número del byte, o la dirección del dígito binario del pixel, se ha de multiplicar el número de carácter **C** por ocho y añadirle el número de fila y 7168, comienzo de la memoria de pantalla. La variable **T** representa el número del byte:

T = 7168 8*C + R

Evidentemente los cálculos se complican. Una de las aplicaciones de los gráficos en alta resolución es poder visualizar fórmulas y ecuaciones matemáticas. La fórmula de una línea o una curva pueden venir dadas por el programa y sus respectivos valores **x** e **y** convertidos, mediante las fórmulas indicadas, en la representación adecuada que hará aparecer la recta o la curva en nuestra pantalla en alta resolución.

C-128

Al referirnos a algunos modos gráficos del C-64, hemos omitido una de sus capacidades más versátiles y útiles: los sprites. La razón es sencilla, los sprites del C-64 fueron tratados en los números 21, 22 y 23. Allí os explicábamos cómo crear sprites multicolores, ampliarlos, hacerlos múltiples, moverlos por la pantalla, etc. Si en ese caso habíamos de recurrir a copiosas sentencias **POKE** y cálculos más o menos tediosos para encontrar las datas adecuadas, las posiciones de memoria correctas y la perfecta visualización a lo largo de toda la pantalla, los cálculos y la programación se simplifican enormemente en el C-128. Podemos, prácticamente, olvidarnos de la plantilla para crear nuestros sprites, debido a la incorporación de comandos como: **SPRAV, SPRITE, MOVSPR, COLLISION, SSHAPE.** Pero veámoslo sobre un ejemplo. La línea 5 pone el fondo de color negro, la línea 10 nos permite trabajar en alta resolución. Ahora podemos dibujar fácilmente recurriendo a **DRAW, CIRCLE, BOX** y **PAINT.** La línea 15 dibuja un rectángulo en la esquina superior izquierda de la pantalla. Las líneas 20 hasta la 32 dibujan el coche de carreras. Las líneas 35 hasta la 46 dibujan las líneas de carrera y la línea de meta. La línea 47 nos permite transformar, mediante **SSHAPE,** el dibujo del coche en una cadena A$.

El dibujo está almacenado ahora en una cadena, podemos introducir movimiento mediante **SPRSAV.** Las datas se transforman ahora en los sprites uno y dos, ambos idénticos. El comando **SPRITE** activa un sprite específico (numerado de uno a ocho), lo colorea, expande su tamaño si fuera preciso, especifica la prioridad de pantalla y determina el modo gráfico que se ha seleccionado. La prioridad se refiere a si el sprite ha de pasar delante o de-

```
1 REM ******************************
2 REM ******* FORMULA 1 ***********
3 REM ***** COMMODORE MAGAZINE ****
5 COLOR 0,1
10 GRAPHIC 1,1
15 BOX 1,2,2,45,45
20 DRAW 1,17,10 TO 28,10 TO 26,30 TO 19,30 TO 17,10
22 DRAW 1,11,10 TO 15,10 TO 15,18 TO 11,18 TO 11,10
24 DRAW 1,30,10 TO 34,10 TO 34,18 TO 30,18 TO 30,10
26 DRAW 1,11,20 TO 15,20 TO 15,28 TO 11,28 TO 11,20
28 DRAW 1,30,20 TO 34,20 TO 34,28 TO 30,28 TO 30,20
30 DRAW 1,26,28 TO 19,28
32 BOX 1,20,14,26,18,90,1
35 BOX 1,150,35,195,40,90,1
37 BOX 1,150,135,195,140,90,1
40 BOX 1,150,215,195,220,90,1
42 DRAW 1,50,180 TO 300,180
43 DRAW 1,50,180 TO 50,190
44 DRAW 1,300,180 TO 300,190
45 DRAW 1,50,190 TO 300,190
46 CHAR 1,18,23,"META"
47 SSHAPE A$,11,10,34,31
50 SPRSAV A$,1
55 SPRSAV A$,2
60 SPRITE 1,1,7,0,0,0,0
65 SPRITE 2,1,3,0,0,0,0
70 MOVSPR 1,240,0
80 MOVSPR 2,120,0
85 MOVSPR 1,180 # 6
90 MOVSPR 2,180 # 7
95 FOR I=1 TO 5000:NEXT
99 GRAPHIC 0,1
```

Gráficos

```
5 REM ******************************
6 REM ********* PLANETAS **********
7 REM ******************************
10 COLOR 0,1:COLOR 1,2:COLOR 3,1:COLOR 4,1
20 GRAPHIC 1,1
30 FOR S=1 TO 180
40 REM ** DIBUJA ESTRELLAS **
50 SY=INT(RND(0)*159)
60 SX=INT(RND(0)*319)
70 DRAW 1,SX,SY:NEXT S
80 REM ** DIBUJA FONDO ******
90 COLOR 1,6
100 FOR Y=160 TO 199 STEP 3
110 DRAW 1,160,160 TO 0,Y:NEXT Y
120 FOR X=0 TO 319 STEP 10
130 DRAW 1,160,160 TO X,199:NEXT X
140 FOR Y=199 TO 160 STEP -3
150 DRAW 1,319,Y TO 160,160:NEXT Y
160 REM ** DIBUJA LINEAS LATITUD NORTE
170 COLOR 1,10:R=32:XR=9+R
180 FOR Y=30 TO 78 STEP 12
190 CIRCLE 1,158,Y,XR,5
200 R=R/2:XR=XR+R:NEXT Y
210 REM ** DIBUJA LINEAS LATITUD SUR
220 R=2:XR=71-R
230 FOR Y=90 TO 126 STEP 12
240 CIRCLE 1,158,Y,XR,5
250 R=R*2:XR=XR-R:NEXT Y
260 REM ** DIBUJA LINEAS LONGITUDINALES
270 R=2:XR=72-R:FOR I=1 TO 8
280 CIRCLE 1,158,78,XR,62
290 R=R+2:XR=XR-R:NEXT I
300 REM DIBUJA ESTRELLA
310 COLOR 1,7
320 CIRCLE 1,40,30,20,18
330 PAINT 1,40,30,1
```

trás de los objetos en la pantalla. El tamaño de los sprites se pueden expandir al doble de su tamaño normal, tanto en sentido vertical como horizontal. El modo gráfico determina si el sprite es un sprite en alta resolución o multicolor. Expliquemos un poco más esto:

SPRITE #, O, C, P, X, Y, M

- **#** Número del sprite (desde uno hasta ocho).
- **O** Encendido (O = 1) o apagado (O = O).
- **C** Color (de 1 a 16).
- **P** Prioridad. Si P = O, el sprite aparece delante de los objetos de la pantalla. Si P = 1, el sprite está detrás de los objetos de la pantalla.
- **X** Si X = 1, espansión horizontal. Si X = O, tamaño normal del sprite.
- **Y** Si Y = 1, expansión vertical. Si Y = O, tamaño normal del sprite.
- **M** Si M = 1, el sprite aparece en el modo multicolor en alta resolución.
  Si M = 0, el sprite aparece en el modo standard en alta resolución.

Hemos conseguido situar al sprite en pantalla. Lo único que debemos hacer es moverlo. Mediante **MOVSPR** controlamos su movimiento y animación. Lo podemos utilizar de dos formas. Primero, lo que nos permite es situar un sprite en pantalla, mediante las correspondientes coordenadas horizontales y verticales. La línea 70 sitúa el sprite uno en la columna 240, fila cero. La línea 80 sitúa el sprite 2 en la columna 120, fila cero. También podemos recurrir a **MOVSPR** para mover los sprites de sus posiciones originales. En nuestra peculiar carrera en las líneas 85 y 90 lo que hacemos es mover los sprite 1 y 2 desde la parte superior a la inferior. El primer número es el número de sprite, el segundo es el número de grados a mover, en relación a la posición original del sprite. El signo # significa que el sprite se mueve en relación a la posición de partida, en vez de a la posición de las líneas 70 y 80. El número final especifica la velocidad del movimiento.

Si ejecutamos el programa lo que obtendremos en pantalla será la carrera entre dos coches. Pero el programa es una mera ayuda. Trata de dibujar otros coches y objetos. Dibuja otros sprites e introdúcelos en la carrera.

En el presente artículo hemos tratado, de forma somera, los gráficos en general. No hemos acudido a explicar las ampliaciones existentes del BASIC que permiten la incorporación de comandos gráficos específicos, ni a detallar la utilización de las tabletas gráficas. El tema de gráficos es tan extenso que os emplazamos a un próximo tratamiento en profundidad del mismo. De momento, y como aperitivo, os incluimos el programa Planetas.

SU PROGRAMA PARA CUALQUIER SISTEMA COMMODORE PUEDE HACERLE GANAR 5.000 PTAS.

EL PRESENTE CONCURSO ESTA ABIERTO A TODOS NUESTROS LECTORES Y SU PARTICIPACION E INSCRIPCION ES GRATUITA. LEA LAS BASES DEL CONCURSO

- NO SE ESTABLECEN LIMITACIONES EN CUANTO A EXTENSION, TEMA ELEGIDO O MODELO DE ORDENADOR
- LOS CONCURSANTES DEBERAN ENVIARNOS A LA DIRECCION QUE FIGURA AL PIE, EL CASSETTE O DISKETTE CONTENIENDO EL PROGRAMA, UNA EXPLICACION DEL MISMO Y, AL SER POSIBLE, UN LISTADO EN PAPEL DE IMPRESORA. SE PODRAN ENVIAR TANTOS PROGRAMAS COMO SE DESEE
- LOS PROGRAMAS, PREVIA SELECCION, SERAN PUBLICADOS EN LA REVISTA, OBTENIENDO TODOS ELLOS 5.000 PTAS.
- LA DECISION SOBRE LA PUBLICACION O NO DE UN PROGRAMA CORRESPONDE UNICAMENTE AL JURADO NOMBRADO AL EFECTO POR "COMMODORE MAGAZINE", SIENDO SU FALLO INAPELABLE
- LOS CRITERIOS DE SELECCION SE BASARAN EN LA CREATIVIDAD DEL TEMA ELEGIDO Y LA ORIGINALIDAD Y/O SENCILLEZ EN EL METODO DE PROGRAMACION GLOBAL
- ENVIAR A: CONCURSO COMMODORE MAGAZINE

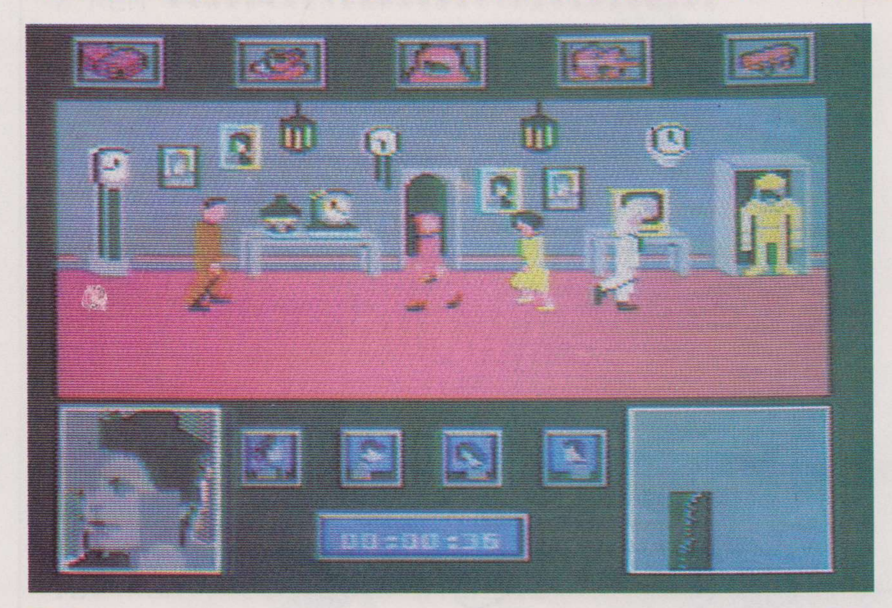


commodore *Magazine*

C/BRAVO MURILLO, 377 5.° A 28020 MADRID



THE COMET GAME



Hace tiempo, los más audaces se preguntaban: ¿hay vida en Marte? Pues bien, ahora parece que el rojo planeta y sus misterios han dado paso a otro cuerpo interestelar mucho más llamativo y desconocido, y que deja tras de sí, además de una bonita historia, una larga y plateada estela sobre la que se ha hablado mucho.

Supongo que ya te imaginas que este nuevo juego de Firebird, trata sobre el Cometa Halley. Un grupo de científicos asegura que dadas las condiciones de temperatura, materiales, etc., que tiene la cola del Cometa Halley, no es imposible que ciertas formas de vida puedan desarrollarse en tan especial lugar. Otros entendidos en la materia se preguntan cómo podría existir algo vivo en un enorme trozo de hielo que viaja por el espacio a una velocidad increíble, y se muestran bastante incrédulos ante tal posibilidad.

Pero queda la duda, y hay que comprobar cuál de las dos hipótesis es cierta. Además, hay otro aspecto a tener en consideración y es que de darse algún tipo de actividad biológica en el Cometa Halley, se trataría de un tipo de bacterias que con toda probabilidad serían un peligro para la vida en la Tierra.

Así pues, habrá que investigar sobre todo ello, ver si realmente existen esas bacterias tóxicas, y en ese caso destruirlas.

Despegamos con nuestra nave hacia una misión realmente importante. Nuestro vehículo espacial dispone de uno de los más modernos sistemas de computadoras, que en todo momento nos mantiene informados sobre lo que ocurre en el espacio exterior, así como de las emergencias que puedan presentarse y que debamos solucionar rápidamente.

Durante el camino hacia el Cometa pueden ocurrir muchas cosas. En el panel de control aparecerán los datos necesarios para mantener la nave en posición correcta hacia nuestro destino. Si la antena de radar se desvía puede ser peligroso, ya que la pérdida del rumbo es una de las mayores dificultades que podemos encontrarnos. Cuando esto ocurra, la computadora de la nave nos avisará y nos dirá también el tiempo máximo que podemos tardar en corregir la posición del radar para alinearlo correctamente.

Otro tipo de emergencias ante las que reaccionaremos con eficacia y rapidez, son las producidas por unos misiles que pueden aparecer en cualquier momento y atacarnos. Si esto ocurre el indicador de protección interplanetaria se activará, avisándonos de que debemos protegernos, pues si los misiles nos alcanzan repetidas veces, nuestra nave puede explotar.

La nave, por supuesto, también está preparada para detectar la presencia de los gérmenes mortales, que supuestamente habitan en el Cometa Halley. Si descubrimos su presencia, no sólo habremos corroborado la primera hipótesis, también nos encontraremos ante la misión de exterminarlos.

Es importante que alcancemos nuestro objetivo. Si así lo hacemos, la humanidad nos estará agradecida.

FICHA DEL JUEGO

NOMBRE:	**THE COMET GAME**
DE:	**FIREBIRD (Serma)**
ORDENADOR:	**COMMODORE 64/128**
CONTROL:	**JOYSTICK**

PUNTUACION	1	2	3	4	5	6	7	8	9	10
ORIGINALIDAD							■			
ADICCION						■				
GRAFICOS								■		
SONIDO						■				
GENERAL							■			

BACK TO THE FUTURE

¿Has visto la última película de Steven Spilberg, «Regreso al futuro»?, si ha sido así, te será muy fácil alcanzar un alto record con este juego, puesto que su argumento está basado en este film.

Un simpático chico llamado Marty, ayudado por un viejo científico loco amigo suyo, consigue transportarse al pasado. Así, casi sin darse cuenta, Marty se encuentra en su misma ciudad con sus mismos edificios y personas, pero 30 años antes

En este «nuevo» ambiente, nuestro amigo conoce a sus propios padres, cuando eran unos jóvenes quinceañeros. El destino estaba preparado para que el padre de Marty, George, se enamorara de Lorraine, la madre de Marty, se casaran y todo discurriera después como Marty lo había conocido en el «futuro». Pero eso de inmiscuirse en el pasado de los progenitores de uno, es más peligroso de lo que nuestro amigo pensó en un principio.

El problema comienza cuando Lorraine, en vez de sentirse atraída por George, se enamora locamente de Marty. Y no sólo eso, además a George le da exactamente igual de lo que pueda hacer Lorraine. Marty no puede permitir que la historia cambie, entre otras cosas, porque si se tuercen los hilos del destino, Marty no nacerá.

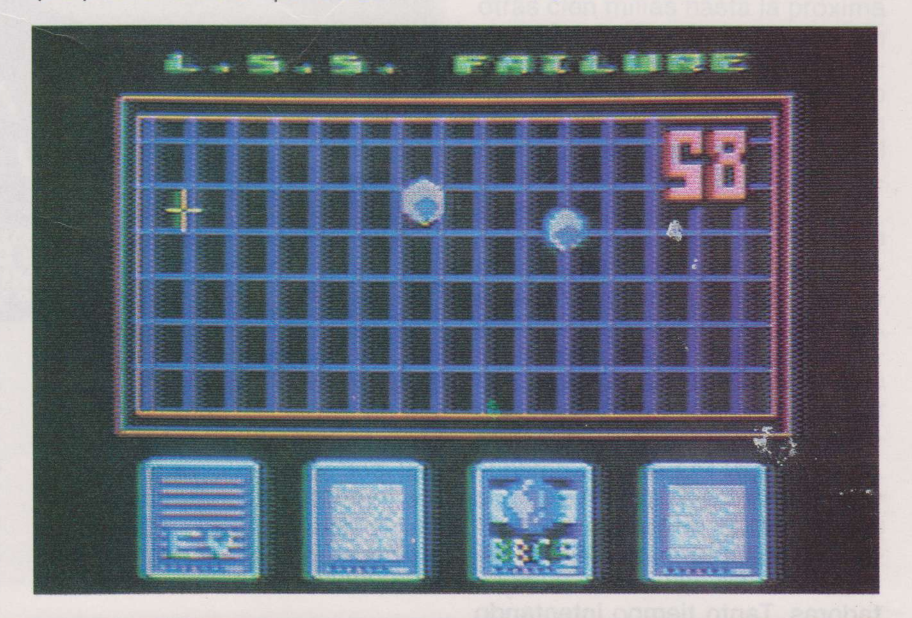
Podemos ir viendo el estado de la situación, por una fotografía de Marty, que verás en la parte inferior izquierda de la pantalla. Cuando las posibilidades de los padres de Marty se enamoran, son bajas, es decir, cuando casi no coincidan, ni se vean, la fotografía irá desapareciendo, poco a poco o por fragmentos.

Marty debe procurar que sus «futuros» padres estén juntos la mayor cantidad de tiempo posible, pero esto es difícil, primero, por la indiferencia que muestran uno hacia otro, y segundo, porque Lorraine no se despega de Marty, sin fijarse siquiera en que George existe.

Además, está el matón del barrio, que procura dificultar la tarea de nuestro amigo, y siempre que puede le da un puñetazo.

Marty también cuenta con ayudas. Su amigo el científico no le ha abandonado, y de vez en cuando le echa una mano. También puede utilizar una serie de objetos que le facilitarán su labor. Hay un poema de amor que Marty debe dar a George para que se lo lea a Lorraine en la cafetería; un traje de extraterrestre, para convencer a George de que vaya al baile con Lorraine. También hay una guitarra, si Marty la encuentra y toca con ella en la escuela de baile, sus padres tendrán más posibilidades de enamorarse.

El juego parece sencillo, pero hace falta algo de práctica para comprenderlo bien y utilizar los objetos en el momento adecuado. Aunque el número de pantallas diferentes que parecen no es muy numeroso, seguro que te divertirás con los graciosos personajes de «Regreso al futuro».

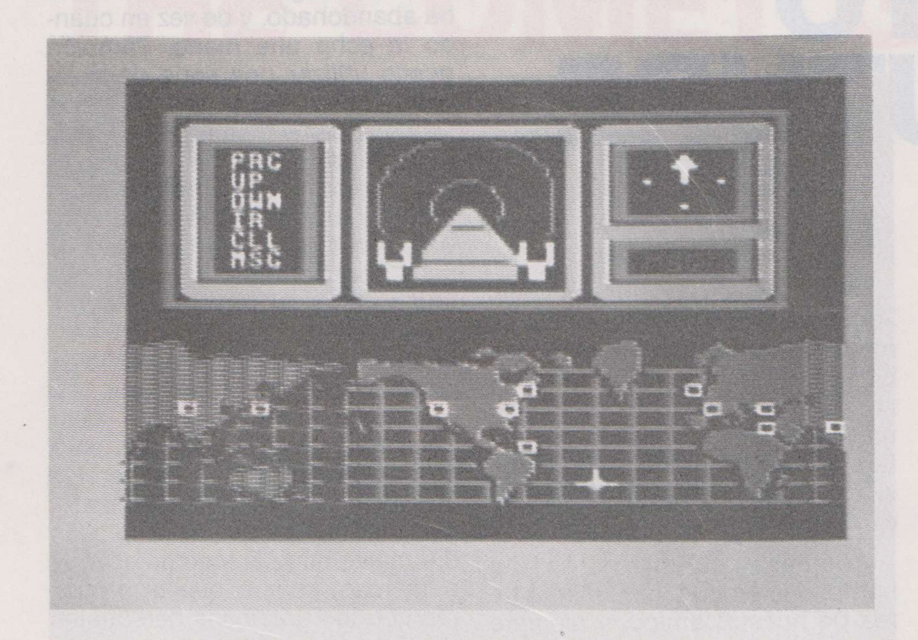


FICHA DEL JUEGO

NOMBRE:	**BACK TO THE FUTURE**
DE:	**ELECTRIC DREAMS (Proeinsa)**
ORDENADOR:	**COMMODORE-64**
CONTROL:	**JOYSTICK/TECLADO**

PUNTUACION	1	2	3	4	5	6	7	8	9	10
ORIGINALIDAD						■				
ADICCION							■			
GRAFICOS							■			
SONIDO								■		
GENERAL							■			





HACKER

«Es de noche, estoy cansada. He tenido un día de trabajo bastante duro ante mi terminal de ordenador, cuando de repente una extraña interferencia aparece en la pantalla. ¿Qué ocurre? ¿Habrá sido un corte de electricidad?... ¿Qué es esto? Se ve un extraño mensaje sobre un fondo azul, LOGON PLEASE... Comencé a pulsar teclas al azar, hasta que me di cuenta de lo que en realidad estaba ocurriendo. Acabo de entrar en otro sistema de computadoras. Tanto tiempo intentando intervenir nuevos sistemas, y ahora así, sin querer he entrado, aún no sé dónde...

El ordenador me pide una serie de claves que ignoro por completo, continúo en el mismo sistema, frases incomprensibles me muestran la clave... y aparece un test que me permite fallar en varias ocasiones hasta que logro completar satisfactoriamente todas las preguntas que me hacen, sobre dos extraños artefactos que han aparecido en la pantalla.

A partir de aquí todo se complica..., quizás nunca debí entrar en este sistema, puede ser demasiado peligroso..., pero no puedo abandonarlo ahora, tengo curiosidad por saber qué va a ocurrir..., me temo que el computador central de este sistema no pertenece precisamente a alguien muy recomendable...»

Hacker es un tipo de juego de simulación, para esas personas que disfrutarían penetrando impunemente en ordenadores ajenos.

El juego no tiene instrucciones, por lo que al cargarlo no sabes qué puede ocurrir. Al principio parece que nunca conseguirás dar con la clave, pero tarde o temprano, cuando menos te lo esperes, te sorprenderás dialogando con no sabes quién por medio de tu ordenador.

A medida que avanza el juego, las cosas se van complicando. Te serán asignados unos números clave que deberás recordar, para posteriormente poder acceder a nuevas informaciones del sistema.

El ordenador te presentará un esquema que representa a un extraño artefacto cuya finalidad desconoces. Si consigues aprenderte de memoria las partes de este aparato, tendrás la posibilidad de acceder a informaciones más confidenciales, y podrás, por fin, saber a quién pertenece el sistema que has interceptado involuntariamente.

No quiero adelantar mucha información sobre lo que puede pasar después, pues el interés del juego reside, precisamente, en la incertidumbre que en todo momento se mantiene.

La nota más característica de Hacker es, sin duda, su originalidad, y también el alto grado de adicción que puede provocar. Con Hacker sabes cómo entrar en el sistema, pero también debes aprender a salir de él.

FICHA DEL JUEGO

NOMBRE:	**HACKER**
DE:	**ACTIVISION (Proeinsa)**
ORDENADOR:	**COMMODORE-64/128**
CONTROL:	**JOYSTICK Y TECLADO**

PUNTUACION	1	2	3	4	5	6	7	8	9	10
ORIGINALIDAD										X
ADICCION								X		
GRAFICOS							X			
SONIDO						X				
GENERAL								X		

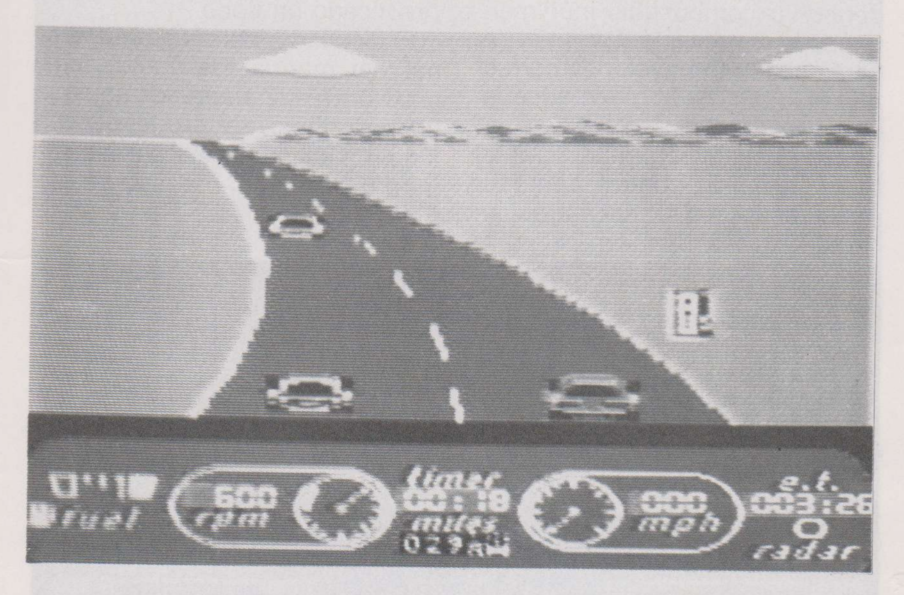
ROAD RACE

Los amantes del riesgo y la velocidad estamos de enhorabuena, porque simplemente cargando Road Race, en nuestro Commodore, podremos disfrutar de una carrera de coches peligrosa y emocionante.

Comienza el juego, presentándonos los preparativos para la carrera, lo primero será conocer a los pilotos que van a competir con nosotros, sus categorías, puntuaciones, etc. Luego seleccionaremos una de las cuatro posibles rutas, en las que podemos correr. Tendremos que procurar estudiar con cuidado el mapa que aparece en pantalla, para escoger la ruta más adecuada a nuestras características, sobre todo si somos principiantes.

Habrá que tener en cuenta factores como las condiciones climatológicas, el tipo de carretera que vamos a escoger y la hora de salida, procurando no coincidir con ninguna hora punta y evitar la conducción durante la noche.

Pues bien, una vez que hemos comprobado todos estos datos y por fin hemos elegido la ruta, podemos comenzar la carrera. Antes de empezarla debemos fijarnos en el panel de instrucciones que aparece en la parte inferior de la pantalla. En él se indican: la cantidad de combustible de que disponemos, el tiempo límite para cubrir la ruta escogida, el indicador de millas, que en todo momento nos dice cuánta distancia nos falta para llegar a nuestro destino, el detector de radar que nos avisa de la proximidad de la policía, el tacómetro, velocímetro...



Podemos pues, comenzar la carrera. Por supuesto, la velocidad es fundamental para poder llegar a tiempo a nuestro destino, pero hemos de tener en cuenta el estado de la carretera y las revoluciones del motor del coche. Puede ocurrir que nuestra forma de conducir sea demasiado brusca y que las ruedas se desgasten, en tal caso tendremos que parar y cambiarlas. Si las revoluciones por minuto son demasiado altas, el motor puede estropearse definitivamente y en tal caso, no tendremos más remedio que empujar el vehículo hasta el garaje más próximo.

Cada cien millas, tendremos la posibilidad de repostar combustible. Aparecerá un anuncio en la pantalla, con la suficiente antelación, como para poder parar en el momento preciso en que lleguemos a la Estación de Servicio. Pero si vamos muy deprisa es posible que no podamos frenar a tiempo y tengamos que esperar otras cien millas hasta la próxima gasolinera. Si nos quedamos sin combustible antes de poder repostar, no habrá más remedio que empujar el coche y confiar en nuestra buena suerte para que el próximo surtidor no esté demasiado lejos.

Bueno, y si después de todo conseguimos llegar a la meta antes de agotar el tiempo, podremos clasificarnos como: ganador, entre los diez primeros o simplemente como merecedores de una calurosa felicitación por haber conseguido llegar a nuestro destino.

Recuerda que para poder clasificarte y grabar tu nombre entre los ganadores, no sólo es necesario la velocidad.

FICHA DEL JUEGO

NOMBRE:	**ROAD RACE**
DE:	**ACTIVISION (Proeinsa)**
ORDENADOR:	**COMMODORE 64/128, ATARI**
CONTROL:	**JOYSTICK/TECLADO**

PUNTUACION	1	2	3	4	5	6	7	8	9	10
ORIGINALIDAD					■					
ADICCION							■			
GRAFICOS						■				
SONIDO						■				
GENERAL						■				



SPACE SHUTTLE

Si lo que realmente te ha gustado siempre de los videojuegos es ese ambiente espacial y futurista que muchas veces tienen, te va a encantar Space Shuttle. Es sin duda, uno de los juegos más completos que hemos visto, que trate sobre un viaje interestelar.

Antes de comenzar la misión propiamente dicha, que es contactar con un satélite en el espacio, debes prepararte para ello. Puedes comenzar con unos vuelos de entrenamiento como los que te propone el autosimulador. Con él te limitarás a observar lo que puede ocurrirte en determinadas situaciones en el vuelo real. Si prefieres practicar de modo que puedas intervenir en los mandos de la nave, como si realmente estuvieras volando, entonces escoge el modo de simulador, en el que la situación es casi la real, pero cuando aún no has despegado.

Todo buen astronauta ha necesitado largas horas de entrenamiento en simulaciones de vuelo, y si tú quieres clasificarte entre los primeros, antes tendrás que practicar un poco.

Bien, cuando consideres que estás suficientemente preparado, lánzate al espacio. Pulsa la tecla «L» y comenzará la cuenta atrás para el despegue. Antes de que el contador llegue a 0 debes poner en marcha el motor primario. Estos y otros muchos datos los encontrarás claramente explicados en un extenso manual de vuelo, que es proporcionado a todos los pilotos.

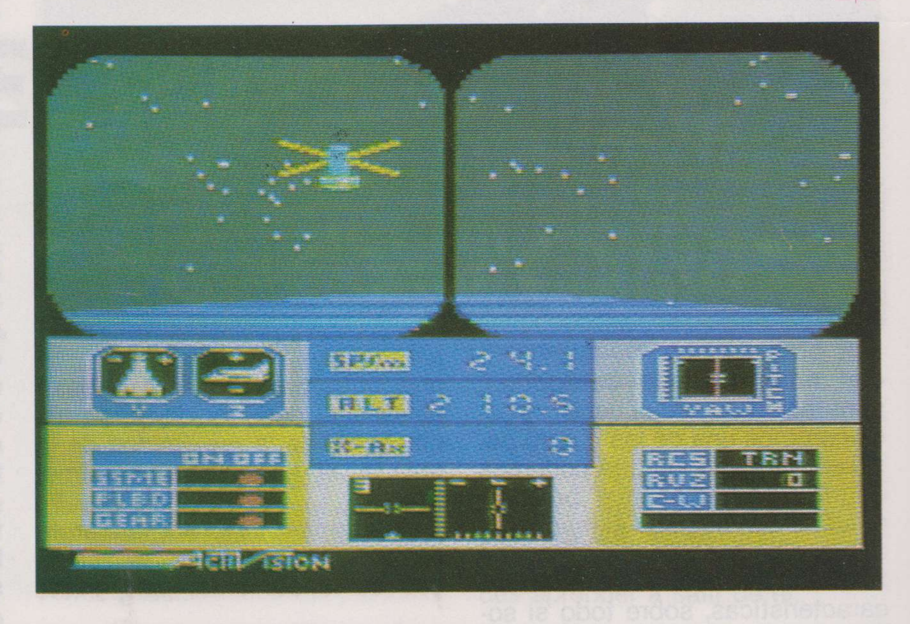
A medida que avances por el espacio, tus objetivos se irán complicando. En un principio, se trata simplemente de despegar e intentar llegar tan cerca de la órbita del satélite como sea posible. Posteriormente deberás conseguir una órbita estable, y ajustar tu velocidad de tal modo que puedas tomar contacto con el satélite.

Los objetivos se complican mucho más, y para realizarlos con éxito, debes estudiar cuidadosamente algunos factores como los que a continuación te sugerimos:

En primer lugar, debes observar continuamente la potencia durante la fase de despegue, para no derrochar combustible inútilmente desde el principio. También deberás seguir la trayectoria correcta y ajustarla en el plano adecuado, moviendo el joystick en las cuatro posiciones. En un completísimo panel de control, podrás ir observando todas tus operaciones, así como las posibilidades que tienes de terminar con éxito tu vuelo.

Respecto a las maniobras de que es capaz el Space Shuttle, siempre que esté guiado por un buen piloto, diremos que se trata de un sistema de maniobras orbital y un sistema de control a reacción. Para poder utilizarlos adecuadamente es necesario haber practicado un poco.

El diseñador del programa, al final del manual de vuelo, hace unos breves comentarios sobre la gran similitud del programa con los simulacros reales que se utilizan en la NASA. Por algo se empieza, ¿no?

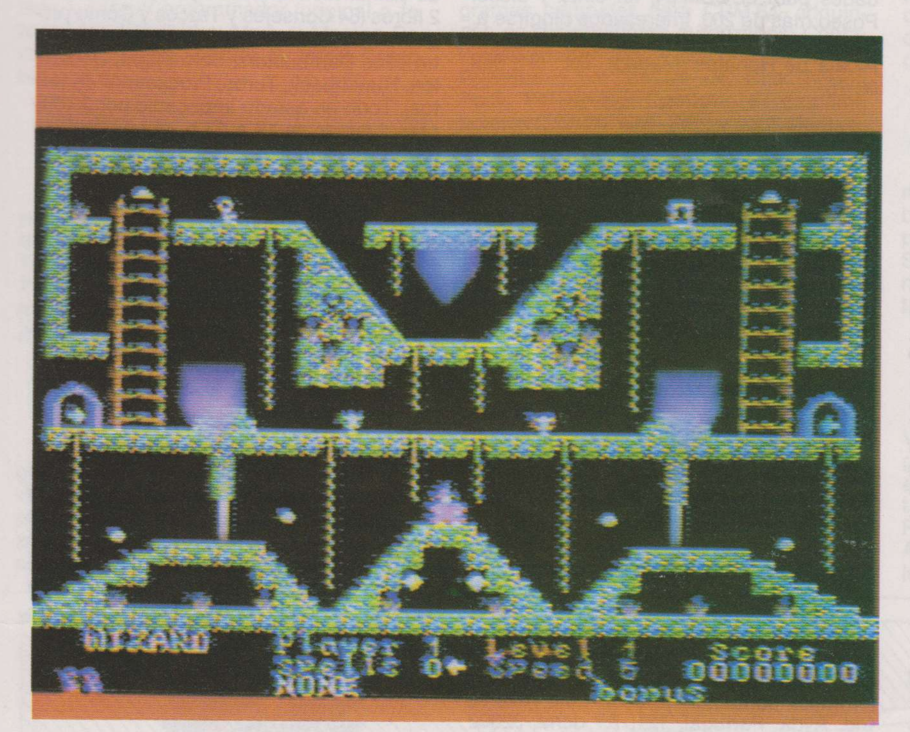


FICHA DEL JUEGO

NOMBRE:	**SPACE SHUTTLE**
DE:	**ACTIVISION (Proeinsa)**
ORDENADOR:	**COMMODORE 64/128**
CONTROL:	**JOYSTICK/TECLADO**

PUNTUACION	1	2	3	4	5	6	7	8	9	10
ORIGINALIDAD						■				
ADICCION								■		
GRAFICOS							■			
SONIDO								■		
GENERAL							■			

WIZARD



Ariolasoft nos presenta un juego realmente «encantado», puesto que Wizard se desarrolla entre magos y hechiceros cuyos secretos intentaremos desvelar.

Estamos ante un juego que se caracteriza fundamentalmente por el gran número de posibilidades que ofrece al usuario. Pueden participar hasta seis jugadores, con cuatro niveles de dificultad, distintas velocidades y opciones de juego. Realmente tardarás en dominarlo por completo.

Wizard se desarrolla en una serie de laberintos, con distintas trampas malévolas, fruto de la magia negra, que debemos evitar.

Al comenzar, aparece nuestro protagonista, el Hechicero surgirá de una nube de humo e inmediatamente se nos informará de la situación, es decir, de qué conjuros somos capaces en un principio y para qué nos servirá cada truco mágico. También sabremos de la cantidad de vidas de que disponemos en cada momento.

El Hechicero tendrá que atravesar diversos niveles del Castillo Mágico para conseguir llegar al tesoro. Durante el camino, encontrará muchos objetos, algunos le servirán de ayuda, mientras que otros, como las hogueras o las calaveras, pueden ser obra de una Magia Negra que intenta exterminarlo.

Para completar un nivel y pasar al siguiente, hay que abrir una puerta encantada, con una llave que previamente tendremos que encontrar.

Nuestro hechicero puede saltar, agacharse, moverse a derecha e izquierda, y por supuesto, lanzar sus conjuros mágicos que, utilizados en el momento oportuno, suelen tener mucho poder.

Te sorprenderá la agilidad con que se mueve el Hechicero. Pero no debe confiarse en ello, porque el Castillo está repleto de trampas y lugares por los que es difícil pasar. Cuando esté sobre las escaleras y cuando tenga que mantenerse en equilibrio sobre las cuerdas, sus pasos habrán de ser más cautelosos.

Puedes ver hasta cuarenta pantallas diferentes de magia, aventuras, trampas y tesoros. Tendrás que atravesar muchos laberintos y enfrentarte a seres muy diversos para conseguir verlas todas. Y si después de todo llegas a ver el tesoro, podrás contarte entre los diez mejores Hechiceros del mundo.

Wizard forma parte de los llamados «juegos de laberintos», caracterizados generalmente por la gran cantidad de pantallas diferentes que se ofrecen, y porque los gráficos suelen ser de tamaño reducido. Esto provoca inevitablemente una pérdida en los detalles de los gráficos, pero también es cierto que agiliza la acción del juego, ya que en cada pantalla ocurren muchas cosas y muy diferentes.

Si eres de los que tienen un buen pulso al manejar el joystick, te divertirás con este juego de Ariolasoft.

FICHA DEL JUEGO

NOMBRE: **WIZARD**
DE: **ARIOLASOFT (Serma)**
ORDENADOR: **COMMODORE-64/128**
CONTROL: **JOYSTICK**

PUNTUACION	1	2	3	4	5	6	7	8	9	10
ORIGINALIDAD							■			
ADICCION					■					
GRAFICOS								■		
SONIDO							■			
GENERAL							■			

¿te interesa?

Desearía intercambiar programas en disco. Tengo un stock de más de 200, entre ellos: Star, Winter Games, A View to a Kill, Exploding Fist, Indiana Jones, Commando, etc., etc. Interesados en el cambio llamar a José M.ª Lladó al teléfono de Granada (958) 22 43 23 o si no escribir a: José M.ª Lladó. C/ Almona del Campillo, 1, 7.° B. 18009 Granada.



Deseo vender máquina de escribir electrónica Brother, a estrenar, con pantalla, dos memorias, conectable a: Commodore, Atari, Sharp como impresora por 38.000 ptas. Ponerse en contacto con el Sr. Barrachina. Tel.: (91) 261 44 49. Madrid.

Vendo Vic-20 y unas cuantas cintas por sólo 8.000 ptas. Ideal para introducirse en la informática. Interesados llamar al tel.: (923) 44 02 10 de Salamanca o bien escribir a: Sebastián Madruga Entisne. C/ Queipo de Llano, 2. La Fuente de S. Esteban. Salamanca.



Si tienes programas en cinta para el C-64 que quieras vender escríbeme enviando lista y las condiciones de venta. Prometo contestar a todos aquellos que me escriban. Mis señas son: Juan Luis Rubio García. Avda. Mariano Andrés, 93, 1.° A. 24008 León.



Vendo dos Commodore 64 + dos *dattassete* + una unidad de discos (1541) + una impresora Seikosha GP 500 VC + un VIC-Switch, en perfecto estado y a un precio de fábula. Se vende todo o si se prefiere por separado. Se incluyen juegos (*cassette* o disco), manuales y libros de utilidad. Llamar al tel.: (93) 664 17 24. Antonio Andreu del Aguila. C/ Pelayo, 4, 1.°. Castelldefels (Barcelona).



Por cambio de ordenador, me interesa una impresora (*Interface* Paralelo Centronics), que cambiaría por el siguiente material: ordenador Vic-20, *cassette* C2N, ampliación de memoria (3K u 8K ó 16K), cartucho ayuda al programador (Vic-1212), Vic-Base 16K (Indescomp), Dibuvic (Indescomp), varias cintas con juegos, así como las revistas Commodore Magazine, Commodore World y Club Commodore (colecciones completas). Dirigirse a: A. Puig Vázquez. C/ Médico Rodríguez, 15, 4.°. La Coruña. Tel.: (981) 25 04 83 de 20,30 a 21,30. También vendo.

Me gustaría intercambiar juegos y utilidades para el CBM-64, en cinta y disco. Poseo más de 200. Interesados dirigirse a: Basilio Puebla Sánchez. C/ Luis Vives, 7. Alcalá de Henares (Madrid). Tel.: (91) 889 66 76.



Desearía intercambiar programas para C-64 preferentemente en disco. Poseo más de 600. En la lista poner el número de bloques que ocupa cada programa. Dirigirse a: Josu Bravo Azkuenaga. C/ Felipe Serrate, 3, 3. D. D. 48013 Bilbao.



Vendo Vic-20 + cartucho lenguaje máquina. Además regalaría la Guía del programador. Interesados llamar a Gerardo al tel.: (948) 82 30 60. Navarra.



Intercambio todo tipo de programas y utilidades para el CBM-64. Tengo más de 250 programas. Interesados enviar lista (prometo contestar) a: José Luis Alvarez Cid. Avda. Portugal, 121, 7.° dcha. 32002 Orense. Enviaré lista.



Por cambio de equipo vendo ordenador Commodore 64, cartucho lenguaje máquina HES-MON 64, todo en perfecto estado con menos de un año. Precio a convenir. Llamar al tel.: (91) 715 41 00 a partir de las 18 horas y preguntar por Díaz.



Deseo ponerme en contacto con usuarios de C-128, para intercambio de experiencias, etc. También cambio programas para CBM-64. Los interesados escribid a: Daniel Roig Marchuet. C/ Castilla, 27, bajos. Ibiza (Baleares). Tel.: (971) 30 07 09 (en horas de comida o cena).



Vendo ordenador Commodore-64 (seminuevo), manuales, cables, etc., por 30.000 ptas. Vendo ordenador Vic-20, en perfecto estado, primer curso de introducción al Basic y cartucho de juegos por 10.000 ptas. Si os interesa podéis llamar o escribir a: Enrique Poveda Trigueros. C/ San Isidro, 28, 5.°. Elche (Alicante). Tel.: (965) 45 06 99, de 14 a 16 horas.

Regalo Commodore 64 (manual, fuente de alimentación, embalaje de fábrica) + 2 libros (64 Consejos y Trucos y Cómo programar su CBM-64 I) + 50 programas (entre ellos Dambusters, Sólo Flight, Decathlon, Hunchback, Turbo, Procesador de textos). Todo en estado impecable por 30.000 ptas. Llamar de 19 a 23 h. al tel.: (91) 215 83 55, o escribir a: Jesús Frías Reyes. C/ Arroyo, 3, 4.° D. 28029 Madrid.



Compro Commodore 64 urgente. Los interesados en vender alguno que escriban a: José Luis Chinchilla Tirado. C/ Cuesta de Maraña, 10, 1.° dcha. 18010 Granada.



Dispongo de más de 500 programas para tu C-64, entre los cuales están los mejores, los últimos aparecidos en USA. No te los pierdas. No los vendo, los intercambio. Decídete y escríbeme a: Vicente Merino de Rueda. C/ Sta. Joaquina de Vedruna, 4, 4.° B. 10001 Cáceres.



Desearía vender Commodore 64 por 35.000 ptas. En el precio van incluidos transformador, cables de conexión al ordenador y algunos juegos importantes como: Rambo, Commando, Summer games II y otros más. También regalaría algunas revistas para Commodore 64 con juegos y utilidades. Interesados llamar al teléfono (958) 27 87 50 o escribir a: José M.ª Lladó Llort. C/ Almona del Campillo, 1, 7.° B. 18009 Granada.



Busco las instrucciones de los siguientes programas: Macro Assembler, Flight Simulator II, Comal 64 y Logo 64. Dirigirse a: Francisco Jesús Gutiérrez Izquierdo. C/ Prolongación Escañuela, 2. 14002 Córdoba.



Urge vender vídeo juego ATARI + 2 *joysticks* + 2 *paddles* + 15 cartuchos. Los programas para CBM-64 Beach Head y LOAD'N'RUN n.° 5, originales. También vendo los números 17 y 18 de Commodore Magazine (150 cada uno). Dirigirse a: Manuel Castaño Cano. Avda. Teodomiro 1, 1.° C. Orihuela (Alicante). Tel.: (956) 30 21 60.

Interesado en cambio de programas para Vic-20, así como comprar guía referencia para el programador (original o fotocopias). Escribid a: Manuel Moreno Romero. Arzobispo Barrios, 2. 14009 Córdoba.



Vendo Vic-20, más *datassette*, más impresora Vic 1515, más muchos programas en cinta. Funcionamiento impecable. Todo junto 30.000 ptas. La impresora sólo 15.000 ptas. Interesados llamar de 2 a 4 h. al tel.: (981) 58 26 45 y preguntar por Alejandro. La Coruña.



Desearía intercambiar todo tipo de juegos y utilidades para Commodore 64. Enviad una lista. Los que estén interesados pueden escribir a: Juan Carlos Sala González. C/ Alejandro Bell, 246. Terrassa (Barcelona). Tel.: (93) 788 37 29.

Por cambio de equipo, vendo, a mitad de precio, negociables, los siguientes programas originales para CBM-64, adquiridos en distribuidores autorizados: Contabilidad 64 (cartucho y disco), Easy Script, Superbase 64, Master 64, Disk Bonus Pack, Pet Speed, Quinielas 64 (imprime boletos CPA 80, New Print y equivalentes), Quinielas 64 (imprime boletos STAR GEMINI), International Soccer (cartucho), Baloncesto. Ponerse en contacto con Rafael Torró. Tel.: (967) 25 01 56, horas comida. Albacete.



Cambio 150 programas para el Commodore 64 por unidad de discos 1541 o por impresora. Todos son comerciales, tanto juegos como utilidades. Ejemplos: Simon's Basic I, S. B. II, Koala Painter, 80 Columnas, Break Dance, Logo, Forth, Pascal, Pitfall II, Contabilidad, Denis II... Interesados llamar a: (947) 23 86 39, en horas de comida, y preguntar por José Miguel.

Desearía vender Commodore 64 con 100 juegos de los últimos que hay en el mercado por sólo 35.000 ptas. Interesados escribir a: José M.ª Lladó Llort. C/ Almona del Campillo, 1, 7.º B. 18009 Granada.



Poseo los siguientes programas: Hesmon-64, Busicalc, Extramon, Logo, Baseball, y los cambiaría por los siguientes programas: Skyfox, Karateka, Broad Street, Misión Imposible, Dambusters. Dirigirse a: Francisco Carnicer Aixala. Apartado Correos, 77. Vendrell (Tarragona).



Somos el mejor club de Commodore de Valencia. Disponemos de más de 400 programas de todo tipo tanto en disco como en cinta. Si deseas contactar con nosotros escríbenos a la siguiente dirección: Radio Club Burjassot Secc. Commodore. Apartado de Correos, 73. Burjassot (Valencia).



Cambio de base

Los amantes del código máquina habrán visto que en los libros sobre el tema, tanto los códigos como los datos se expresan habitualmente en hexadecimal (B_{16}). Lamentablemente, el Basic del Commodore no posee ningún comando que de forma directa pueda realizar estos cambios de bases.

Este truco que os proponemos, no sólo permite pasar un número decimal a hexadecimal o binario, ejemplos para los cuales ya han sido publicadas rutinas en anteriores números, sino que puedes elegir cualquier base comprendida entre 2 y 36. Para representar dígitos mayores de 9 se utiliza el alfabeto, siendo A = 10, B = 11... Z = 36.

El método utilizado es el habitual para cualquier base. Consiste en realizar divisiones sucesivas del número a convertir por la base, hasta que el cociente sea 0. De este modo los restos que vamos obteniendo componen los dígitos que forman el resultado.

```
40 PRINT"♥"
50 INPUT"NUMERO A CONVERTIR";N
60 IF INT(N)<>N THEN 50
70 INPUT"BASE (2-36)";B
80 IF B<2 OR B>36 THEN 70
90 IF INT(B)<>B THEN 70
100 R=1:R$=""
110 NUM$="0123456789ABCDEFGHIJ"
115 NUM$=NUM$+"KLMNOPQRSTUVWXYZ"
120 R=INT(N/B)
130 P=N-B*R
140 R$=MID$(NUM$,P+1,1)+R$
150 N=R
160 IF R>0 THEN 120
170 PRINT"RESULTADO: ";R$
180 PRINT
190 GOTO 50
```

Un uso posible es convertir a hexadecimal el contenido de una dirección de memoria cualquiera examinado mediante el comando PEEK, para lo cual deberéis suprimir las líneas 50, 180 y 190. En la línea 170 suprimir su contenido y colocar un RETURN. El número a convertir hay que meterle en la variable **n**, y ya como último paso habrá que colocar en cualquier línea del programa un GOSUB 60 para ir obteniendo sucesivos valores que se almacenarán en la variable **r$**.

Cambia la forma del cursor

Cuando se piensa en el cursor fácilmente le vemos como un objeto parpadeante, al que podemos mover a través de la pantalla. Sin embargo, no existe un carácter específico que haga de cursor. Con este programa podrás personalizar el cursor dándole la forma de un guión de subrayado.

En el Commodore 64 se utilizan dos posiciones de memoria (211 y 214) para controlar la posición del cursor. Cualquier carácter colocado en estas posiciones se mostrarán de forma intermitente debido a que el ordenador lo va presentando alternativamente en el modo normal o en vídeo inverso.

Esto nos indica que para cambiar la forma del cursor tenemos que redefinir todos los caracteres, de forma que en lugar de aparecer el inverso aparezcan caracteres subrayados.

Aquí tienes el programa que realiza esto.

```
10 POKE 55,0:POKE 56,56 :CLR
20 POKE 56334,0:POKE 1,51
30 FORA=0TO1023:POKEA+14336,PEEK(A+53248)
40 POKEA+15360,PEEK(A+53248):NEXT
50 FOR A=15367 TO A+1023STEP8
60 POKE A,255:NEXT: POKE1,55
70 POKE 56334,1:POKE 53272,30
```

Después de unos 30 segundos, tiempo que emplean los POKEs para almacenar el nuevo conjunto de caracteres en memoria (a partir de la posición 14336), el cursor aparecerá como un guión de subrayar intermitente. Con este programa la opción de vídeo inverso del Commodore desaparece y en su lugar, si pulsas CTRL-9 (RVS ON), las letras aparecerán subrayadas.

Eliminograma



Los juegos de inteligencia, también llamados «pasatiempos», gozan de una marcada preferencia entre el gusto de nuestros lectores. Esto es debido, al menos así lo creemos, a su facultad de enseñar mientras divierten y entretienen. Pensamos que estas dos características han sido tomadas en cuenta por Fernando Cruz en la realización del programa denominado Eliminograma que para la sección de concurso nos ha remitido desde Alicante.

Este programa está realizado para el C-64 y su objetivo final es adivinar una palabra de cinco letras pensada previamente por el ordenador. Para ello podrás hacer uso de un cuadro situado en la parte izquierda de la pantalla, en el que hay impresas 10 palabras distintas de 5 letras cada una. Al lado de cada una de estas pala-letras hasta conseguir una palabra conocida y lógica. Pulsando F7 introduciremos la solución, que si no es correcta habrá que seguir buscando más combinaciones lógicas de esas 5 letras y volver a pulsar F7.

Si con la práctica llegas a tal punto que el adivinar la palabra no te resulta complicado, siempre tienes el reto de adivinarla en el menor tiempo posible, controlado desde el inicio del juego por un reloj situado en la parte inferior del recuadro.

Ahora dos recomendaciones a la hora de teclear el programa:

— hay que poner especial cuidado al teclear las líneas

PROGRAMAS

400 y 410, pues si no, el programa no funcionará bien.
— si se quiere poner más o menos palabras en las DATAs habrá que modificar el número 96 de la línea 40.
bras aparece un número, que representa letras de esa palabra que a su vez forman parte de la palabra que hay que adivinar.

Ejemplo: una de las palabras dispuestas en el cuadro es "rosal 2". El 2 indica que dos letras de esta palabra forman parte de la que hay que adivinar (en este caso la R y la O de la palabra "Ratón").

Con el joystick irás seleccionando las letras, eliminando con la barra espaciadora o el botón del joystick todas aquellas que no te sean útiles. Si deseas recuperar alguna letra borrada por equivocación pulsa otra vez el disparador del joystick.

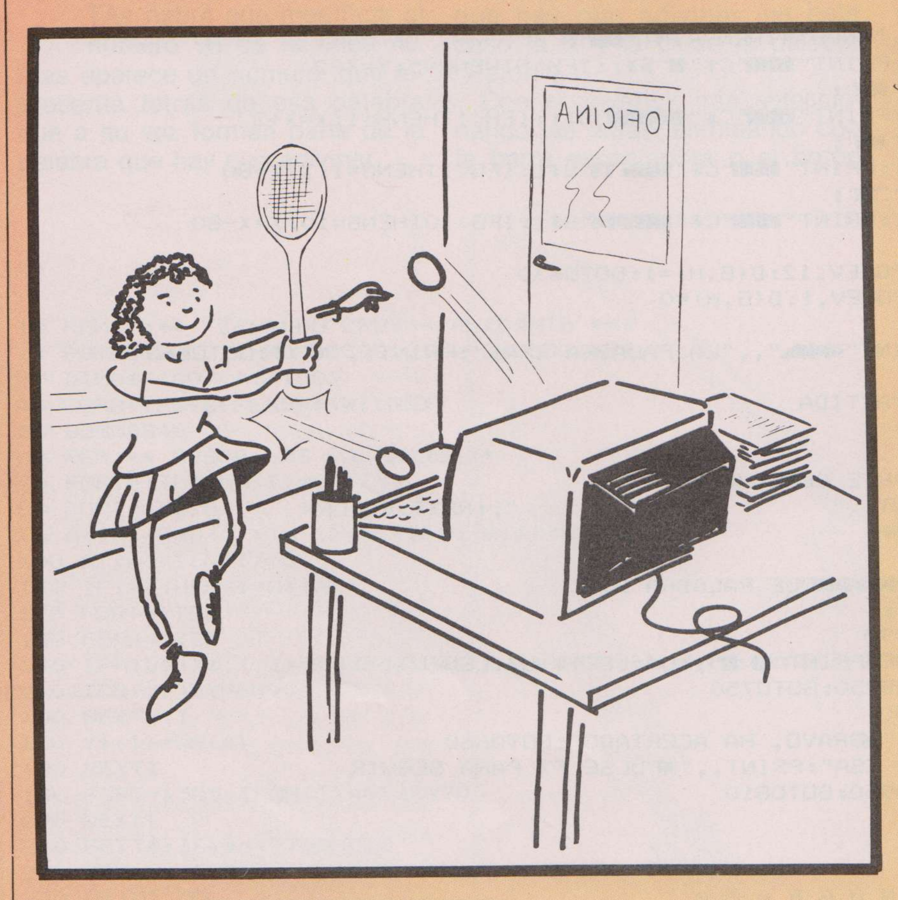
En cuanto creas saber, por medio de la eliminación, las 5 letras que componen la palabra a descubrir, habrá que hacer mentalmente combinaciones con esas 5

CONCURSO
PREMIADO CON
5.000
PESETAS

```
10 REM  *** FERNANDO CRUZ - ALICANTE ***
20 POKE53280,11:POKE53281,11:POKE650,128
30 DIMW$(100),A1(100)
40 FORW=1TO96:READW$(W):NEXT
50 GOSUB840
60 REM ** BUSCA LAS PALABRAS **
70 FORT=1TO5:MM(T)=0:NEXT
80 FORI=0TO10
90 A=INT(RND(A)*(W-1)+1):IFA1(A)=1THEN90
100 A1(A)=1:N(I)=0
110 IFI=0THENX$=W$(A)
120 FORT=1TO5
130 FORJ=1TO5
140 IFMID$(X$,T,1)=MID$(W$(A),J,1)THENN(I)=N(I)+1:IFI>0THENMM(T)=1
150 IFN(0)>5THEN90
160 NEXTJ,T
170 Y$(I)=W$(A)
180 NEXTI
190 FORT=1TO5:IFMM(T)=0THEN70
200 NEXTT
210 GETT$:IFT$=""THEN210
220 REM ** IMPRIME LAS PALABRAS Y OPCIONES **
230 PRINT"▒▒___________________________"
240 PRINT"| "TAB(18)" |"
250 FORI=1TO10
260 PRINT"| ▒▒";:FORJ=1TO5:PRINTMID$(Y$(I),J,1)" ";:NEXT:PRINT"▒ = "N(I)"▒ |"
270 PRINT"| "TAB(18)" |"
280 NEXT
290 PRINT"‾‾‾‾‾‾‾‾‾‾‾‾‾‾‾‾‾‾‾‾‾‾‾‾‾‾‾"
300 SYS65499
310 PRINT"▒▒",,"▒    P U L S E
320 PRINT,,"▒'A' =  ARRIBA
330 PRINT,,"▒'Z' =  ABAJO
340 PRINT,,"▒'<' =  IZQDA.
350 PRINT,,"▒'>' =  DERECHA
360 PRINT,,"▒ESPACIO = BORRAR
370 PRINT,,"▒D JOYSTICK EN PORT 2
380 PRINT,,"▒F1 PARA RENDIRSE
390 PRINT,,"▒F7 DAR LA SOLUCION
400 S$="▒▒ ▒▒└─┘▒▒▒▒┌─┐▒▒| ":X=1106:PRINT"▒"
410 C$=" ▒▒   ▒▒▒▒   ▒▒ "
420 PRINT"▒▒"S$;:H=1:G=1
430 REM ** BUCLE PRINCIPAL **
440 POKE198,0
450 GETT$:O=PEEK(56320)
```

```
460 IFO=126THENT$="A"
470 IFO=125THENT$="Z"
480 IFO=119THENT$="."
490 IFO=123THENT$=","
500 IFO=111THENT$=" "
510 GOSUB950:IFT$=""THEN450
520 FORJ=1TO120:NEXT:IFT$="."ANDH>4THENPRINT"";
530 IFT$="."THENX=X+2:H=H+1:PRINT""C$""S$;:IFH>5THENH=5:X=X-2
540 IFT$=","ANDH<2THENPRINT"";
550 IFT$=","THENX=X-2:H=H-1:PRINT""C$""S$;:IFH<1THENH=1:X=X+2
560 IFT$="A"ANDG<2THENPRINT"";
570 IFT$="A"THENX=X-80:G=G-1:PRINT""C$""S$;:IFG<1THENG=1:X=X+80
580 IFT$="Z"ANDG>9THENPRINT"";
590 IFT$="Z"THENX=X+80:G=G+1:PRINT""C$""S$;:IFG>10THENG=10:X=X-80
600 V=54272+X
610 IFT$=" "ANDD(G,H)=0THENPOKEV,12:D(G,H)=1:GOTO630
620 IFT$=" "ANDD(G,H)=1THENPOKEV,1:D(G,H)=0
630 IFT$=""THEN720
640 IFT$=""THENGOSUB700:PRINT"",,"LA PALABRA ERA:":PRINT,,""X$:GOTO660
650 GOTO440
660 PRINT,,"F7 PARA OTRA PARTIDA
670 GETT$:IFT$<>""THEN670
680 RUN
690 REM ** RUTINA BORRADO PARTE DERECHA **
700 PRINT"":FORT=1TO20:PRINT,,"                    ";:NEXT:RETURN
710 REM ** DAR LA SOLUCION **
720 PRINT""C$
730 GOSUB700:PRINT"",,"QUE PALABRA ES?
740 U$="":PRINT:PRINTTAB(23);
750 GETT$:IFT$=CHR$(13)THEN790
760 IFT$=CHR$(20)ANDU$<>""THENPRINT"  ";:U$=LEFT$(U$,LEN(U$)-1)
770 IFT$<"A"ORT$>"Z"THENGOSUB950:GOTO750
780 PRINTT$;:U$=U$+T$:GOTO750
790 PRINT:IFU$=X$THENPRINT,,"BRAVO, HA ACERTADO":GOTO660
800 PRINT,,"LO SIENTO, NO ES ESA":PRINT,,"PULSE F7 PARA SEGUIR
810 GETT$:IFT$<>""THENGOSUB950:GOTO810
820 GOSUB700:GOTO310
830 REM ** PRESENTACION **
840 PRINT""TAB(6)"                      "
850 PRINTTAB(6)" E L I M I N O G R A M A "
860 PRINTTAB(6)"                         "
870 PRINT"MEDIANTE UN SISTEMA DE ELIMINACION,
880 PRINT"TENDRA QUE ADIVINAR UNA PALABRA DE 5
890 PRINT"LETRAS, TENIENDO EN CUENTA QUE LA CIFRA
900 PRINT"DE LA DERECHA INDICA EL NUMERO DE LETRAS
910 PRINT"IGUALES QUE HAY EN LA PALABRA A BUSCAR
920 PRINT,"PULSA UNA TECLA
930 RETURN
940 REM ** RELOJ **
950 M$=MID$(TI$,3,1):M1$=MID$(TI$,4,1)
960 M2$=MID$(TI$,5,1):M3$=RIGHT$(TI$,1)
970 POKE1944,VAL(M$)+48:POKE1945,VAL(M1$)+48
980 POKE1946,58:POKE1947,VAL(M2$)+48:POKE1948,VAL(M3$)+48
990 RETURN
1000 DATAARADO,NACAR,NARDO,NORTE,OASIS,SALON,TRAMO,MELON,CLAVO,MIRLO,RELOJ,LISTA
1010 DATAJAULA,BLUSA,FELIZ,PATIN,REGAR,CUTIS,LIMON,PIANO,IDOLO,ROCIO,SOLAR,CUEVA
1020 DATATURCO,LUNES,REZAR,TUMOR,TAZON,BARCA,CALIZ,TRACA,CREMA,ACERO,DEDAL,TRONO
1030 DATAOPERA,PLATO,HACHA,HIELO,HUEVO,HIENA,FUMAR,CIELO,VALOR,SALUD,TROPA,GORDO
1040 DATATRIGO,FRESA,RUEDA,OREJA,BURRO,TORPE,MIEDO,PLOMO,SUELO,FECHA,FLOJO,VIEJO
1050 DATAJOVEN,CABRA,DUCHA,FIRMA,LINEA,CURVA,NUEVO,BRAZO,CLAVO,PRIMO,LAPIZ,GRANO
1060 DATAJAMON,CALOR,CIEGO,MANTA,BIDON,TINTA,TIGRE,NIEVE,MUELA,LICOR,AVENA,PRADO
1070 DATAROSAL,PALCO,ALETA,PRESO,SELLO,SUDOR,VAPOR,HUMOR,POLVO,TRIBU,CAJON,CABLE
```

Tenis C-64



Para que no todo sea trabajo y dado que siempre después de una pausa se retoma el estudio con más ímpetu, te invitamos a que te tomes un descanso de la mano del programa que a continuación te ofrecemos.

Aunque el programa simula un partido de tenis, éste no te resultará agotador, pues sentado en un plácido sillón todo tipo de deporte resulta más llevadero.

Como ya he mencionado, se trata de un juego de tenis en el que podrán participar dos jugadores. El juego en sí, es facil de manejar. Una vez tecleado (prestando gran atención al teclear las líneas DATAs) y ejecutado el programa aparece una pantalla de presentación con el mensaje Tenis, que permanecerá en pantalla mientras se cargan los datos necesarios. A continuación, habrá que introducir los nombres de los participantes. El primero jugará en la parte izquierda de la pista y utilizará las teclas Commodore y Shift para el desplazamiento. El otro jugador se situará en la parte derecha y con las teclas de función F5 y F7 podrá desplazarse. El tanteo que se vaya obteniendo en cada jugada se muestra simultáneamente en un marcador electrónico situado en la parte superior de la pantalla.

El juego es rápido, pero si queréis más o menos rapidez, no tenéis más que modificar el data de la línea 11245. El autor nos recomienda que el valor de esta data siempre esté comprendido entre 252 y 255.

También podéis cambiar el aspecto de los jugadores, modificando los datos de las líneas 9200 a 9310.

Estructura del programa:

10-140	Pantalla presentación.
150-180	Bucle para los datos.
190-240	Posición y color de los Sprites.
260	Inicializa y pone a cero las variables de puntuación.
270	Llamada a la rutina en C.M. de pantalla y color del borde.
280-330	Coloca los Sprites en la pantalla.
335-336	Variables de puntuación.
337	Primera jugada
340-440	Inicio de posteriores jugadas.
410-550	Controlan el marcador.
1000-1100	Mueve el jugador que gana el set.
2000-2040	Pide seguir jugando o no cuando finaliza.
4000-4020	Escribe el marcador.
5000-5055	Pide iniciales del jugador.
9000-9310	Datas sprites.
10000-10200	Datas C.M. pantalla.
11000-11260	Datas C.M. juego.

Las rutinas de código máquina son:

SYS 49152	Dibuja pantalla.
SYS 49664	Mueve la pelota.
SYS 49733	Mueve la pelota.
SYS 49802	Mueve la pelota.
SYS 49870	Mueve la pelota.

```
10 PRINT"▣":PRINT:PRINT:PRINT:PRINT
20 POKE53281,7:POKE53280,7
30 PRINT"▥   ▚      ■  ▚      ■  ▚ ■  ▚ ■  ▚ ■  ▚      ■'
40 PRINT"▥   ▚      ■  ▚      ■  ▚ ◥■ ▚ ■  ▚ ■  ▚      ■'
50 PRINT"▥   ▚      ■  ▚  ■     ▚  ◥▚  ■  ▚ ■  ▚   ■'
60 PRINT"▥     ▚ ■     ▚    ■   ▚       ■  ▚ ■  ▚     ■'
70 PRINT"▥     ▚ ■     ▚    ■   ▚ ■◥▚  ■  ▚ ■  ▚      ■'
80 PRINT"▥     ▚ ■     ▚  ■     ▚ ■ ◥▚ ■  ▚ ■       ▚ ■'
90 PRINT"▥     ▚ ■     ▚      ■  ▚ ■  ▚ ■  ▚ ■  ▚      ■'
100 PRINT"▥     ▚ ■     ▚      ■  ▚ ■  ▚ ■  ▚ ■  ▚      ■'
110 PRINT:PRINT:PRINT:PRINT
120 PRINT"                    POR    ":PRINT
130 PRINT"                MARIANO  ARIAS   "
140 FORT=1TO500:NEXT
150 POKE53281,0:POKE53280,0
160 FORI=12352TO12607:READA:POKEI,A:NEXT
170 FORI=49152TO49451:READA:POKEI,A:NEXT
180 FORI=49664TO50048:READA:POKEI,A:NEXT
190 V=53248:POKE650,128
200 POKEV+21,255
205 POKE2040,193:POKE2041,196:POKE2042,196
210 POKE2043,194:POKE2044,193
220 POKEV+39,7:POKEV+40,0:POKEV+41,1
230 POKEV+42,6:POKEV+43,6
240 POKEV+23,8
260 GOSUB5000:FORT=1TO6:N1(T)=0:N2(T)=0:NEXT:J1=0:J2=0:G1=0:G2=0
270 SYS49152:FORI=1325TO1965STEP40:POKEI,229:NEXT:POKE53280,6
280 POKEV+16,0
290 POKE781,72:POKE782,170:POKEV,72:POKEV+1,170
300 POKEV+2,67:POKEV+3,170
310 POKEV+16,PEEK(V+16)OR4
320 POKEV+4,25:POKEV+5,170
330 POKEV+6,32:POKEV+7,58
335 P=1:T1(1)=15:T1(2)=30:T1(3)=40:T1(4)=1:T1(5)=2
336 T2(1)=15:T2(2)=30:T2(3)=40:T2(4)=1:T2(5)=2
337 GOSUB4000:SYS49664:GOTO410
340 GOSUB4000
350 Q=INT(RND(1)*2):W=INT(RND(1)*120)+106
360 POKE782,W
370 IFE=5ANDQ=<1THENPOKEV+16,PEEK(V+16)AND254:POKE781,72:SYS49664:GOTO410
380 IFE=5ANDQ=>1THENPOKEV+16,PEEK(V+16)AND254:POKE781,72:SYS49733:GOTO410
390 IFE=4ANDQ=<1THENPOKEV+16,PEEK(V+16)OR5:POKE781,1:SYS49802:GOTO410
400 IFE=4ANDQ=>1THENPOKEV+16,PEEK(V+16)OR5:POKE781,1:SYS49870:GOTO410
410 E=PEEK(V+16)
420 IFE=4THEN500
430 J1=J1+1
440 IFJ1=4ANDJ2<3THENN1(P)=N1(P)+1:GOSUB4000:J1=0:J2=0
450 IFJ1=5ANDJ2=3 THENN1(P)=N1(P)+1:GOSUB4000:J1=0:J2=0
460 IFN1(P)=6ANDN2(P)<=4THENSP=0:G1=G1+1:GOTO1000
470 IFN1(P)=7THENSP=0:G1=G1+1:GOTO1000
475 IFJ1=6THENJ1=3
480 GOTO340
500 J2=J2+1
510 IFJ2=4ANDJ1<3THENN2(P)=N2(P)+1:GOSUB4000:J1=0:J2=0
520 IFJ2=5ANDJ1=3 THENN2(P)=N2(P)+1:GOSUB4000:J1=0:J2=0
530 IFN2(P)=6ANDN1(P)<=4THENSP=1:G2=G2+1:GOTO1000
540 IFN2(P)=7THENSP=1:G2=G2+1:GOTO1000
545 IFJ2=6THENJ2=3
550 GOTO340
1000 GOSUB4000
1010 POKE2041,196:POKE2042,196
```

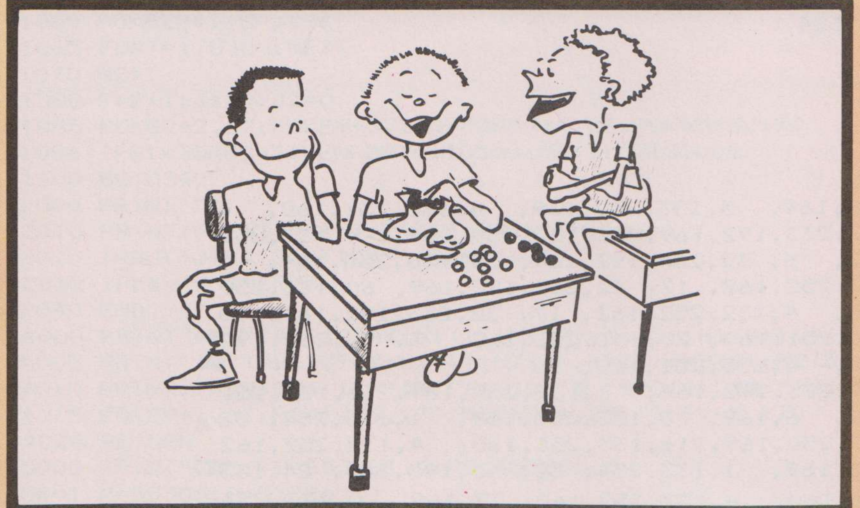
PROGRAMAS

```
1030 FORI=1TO7
1040 POKE2041+SP,195
1050 FORT=1TO300:NEXT
1060 POKE2041+SP,196
1065 FORT=1TO300:NEXT
1070 NEXT
1080 P=P+1:J1=0:J2=0
1085 POKEV+2,67:POKEV+3,170:POKEV+4,25:POKEV+5,170
1086 IFG1=3ORG2=3THENFORT=1TO500:NEXT:GOTO2000
1100 GOTO340
2000 PRINT"■"
2010 PRINT:PRINT:PRINT:PRINT:PRINT:PRINT
2020 INPUT"OTRA PARTIDA? (S/N)";E$
2030 IFE$="S"THEN260
2040 END
4000 PRINT"■■■"TAB(10)A$;N1(1);N1(2);N1(3);N1(4);N1(5);N1(6)
4005 PRINT"■■"TAB(10)B$;N2(1);N2(2);N2(3);N2(4);N2(5);N2(6)
4010 PRINT:PRINTTAB(14)"■■ "T1(J1)"■ "SPC(4)"■■ "T2(J2)"■■ "
4015 POKEV+2,67:POKEV+3,170:POKEV+4,25:POKEV+5,170
4020 RETURN
5000 PRINT"■■":POKE53281,0:POKE53280,0
5001 A=55306:B=55326
5002 FORI=1TO5:FORT=ATOB:POKET,0:NEXTT:A=A+40:B=B+40:NEXTI
5010 PRINT:PRINT:PRINT:PRINT:PRINT:PRINT
5020 PRINT"■INICIALES DEL PRIMER JUGADOR  :  ...■■■■■■";:INPUTA$
5030 IFLEN(A$)>3THENA$=LEFT$(A$,3)
5040 PRINT"■■■■■INICIALES DEL SEGUNDO JUGADOR :  ...■■■■■■";:INPUTB$
5050 IFLEN(B$)>3THENB$=LEFT$(B$,3)
5055 RETURN
8999 REM **** DATAS SPRITES
9000 DATA0,0,0,0,0,0,0
9001 DATA0,0,0,0,0,0,0
9002 DATA0,0,0,30,0,0,63
9003 DATA0,0,63,0,0,63,0
9004 DATA0,63,0,0,30,0,0
9005 DATA0,0,0,0,0,0,0
9006 DATA0,0,0,0,0,0,0
9007 DATA0,0,0,0,0,0,0
9008 DATA0,0,0,0,0,0,0
9010 DATA0
9100 DATA255,255,255,255,143,241,156
9101 DATA12,1,156,24,1,140,31
9102 DATA129,140,56,1,140,48,1
9103 DATA128,48,1,128,48,15,225
9104 DATA63,159,241,0,33,249,6
9105 DATA33,237,0,63,229,6,31
9106 DATA231,12,3,227,24,3,227
9107 DATA24,3,225,24,3,225,25
9108 DATA255,225,17,255,255,255,255
9110 DATA0
9200 DATA0,56,0,56,68,48,68
9201 DATA108,80,68,68,48,68,40
9202 DATA48,40,16,224,17,239,192
9203 DATA19,184,0,63,124,0,24
9204 DATA124,0,16,124,0,16,124
9205 DATA0,0,56,0,0,108,0
9206 DATA0,198,0,1,131,0,1
9207 DATA131,0,1,131,0,0,130
9208 DATA0,0,68,0,1,198,0
9210 DATA0
9300 DATA0,56,0,0,68,0,56
```



```
9301 DATA108,0,68,68,0,68,40
9302 DATA0,68,16,0,40,239,0
9303 DATA17,185,128,19,124,192,62
9304 DATA124,96,24,124,96,16,124
9305 DATA64,16,56,0,0,108,0
9306 DATA0,198,0,1,131,0,1
9307 DATA131,0,1,131,0,0,130
9308 DATA0,0,68,0,1,199,0
9310 DATA0
9999 REM ***** DATAS GRAFICOS
10000 DATA169,  7,141, 33,208,169,  5,133,250,169,  4,133,251,160,  4
10010 DATA132,252,162,  6, 32,233,192,169, 31,133,250,169,  4,133,251
10020 DATA160,  4,132,252,162,  6, 32,233,192,169, 24,133,250,169,  5
10030 DATA133,251,160,  4,132,252,162, 17, 32,233,192,169, 60,133,250
10040 DATA169,  5,133,251,160,  4,132,252,162, 17, 32,233,192,169, 10
10050 DATA133,250,169,  4,133,251,160, 20,132,252,162,  5, 32,233,192
10060 DATA169, 29,133,250,169,  5,133,251,160, 30
10070 DATA132,252,162, 17, 32,233,192,169,  5,133,250,169,216,133,251
10080 DATA160,  4,132,252,162,  6,169,  0,133,253,169,  1,133,254, 32
10090 DATA  3,193,169, 31,133,250,169,216,133,251,160,  4,132,252,162
10100 DATA  6,169,  0,133,253,169,  1,133,254, 32,  3,193,169, 24,133
10110 DATA250,169,217,133,251,160,  4,132,252,162, 17,169,  0,133,253
10120 DATA169,  1,133,254, 32,  3,193,169, 60,133,250,169,217,133,251
10130 DATA160,  4,132,252,162, 17,169,  0,133,253
10140 DATA169,  1,133,254, 32,  3,193,169, 29,133,250,169,217,133,251
10150 DATA160, 30,132,252,162, 17,169,  5,133,253,169,  0,133,254, 32
10160 DATA  3,193, 96,160,  0,169,160,145,250,200,196,252,208,249,165
10170 DATA250, 24,105, 40,144,  2,230,251,133,250,202,208,231, 96,160
10180 DATA  0,165,253,145,250, 24,101,254,201, 15,240,245,200,196,252
10190 DATA208,242,168,165,250, 24,105, 40,144,  2,230,251,133,250,152
10200 DATA160,  0,202,208,224, 96,  0,  0,  0,  0
10999 REM ***** DATAS JUEGO
11000 DATA232,142,  0,208,200,140,  1,208, 32, 18,195, 32,110,195,173, 16
11010 DATA208,201,  5,240, 25,224,255,240, 30,192,229,144,  3, 76, 69,194
11020 DATA173, 30,208,201,  5,208,  3, 76,206,194, 76,  0,194, 96,224, 25
11030 DATA208,  1, 96, 76, 25,194, 96,173, 16,208,  9,  1,141, 16,208,162
11040 DATA  0, 76, 25,194, 96,232,142,  0,208,136,140,  1,208, 32, 18,195
11050 DATA 32,110,195,173, 16,208,201,  5,240, 25,224,255,240, 30,192,106
11060 DATA176,  3, 76,  0,194,173, 30,208,201,  5,208,  3, 76,138,194, 76
11070 DATA 69,194, 96,224, 25,208,  1, 96, 76, 94,194, 96,173, 16,208,  9
11080 DATA  1,141, 16,208,162,  0, 76, 94,194
11090 DATA 96,202,142,  0,208,136,140,  1,208, 32, 18,195, 32,110,195,173
11100 DATA 16,208,201,  5,240, 26,224, 48,208,  1, 96,192,106,176,  3, 76
11110 DATA206,194,173, 30,208,201,  3,208,  3, 76, 69,194, 76,138,194, 96
11120 DATA224,  0,240,  3, 76,164,194,173, 16,208, 41,254,141, 16,208,162
11130 DATA255, 76,164,194, 96,202,142,  0,208,200,140,  1,208, 32, 18,195
11140 DATA 32,110,195,173, 16,208,201,  5,240, 26,224, 48,208,  1, 96,192
11150 DATA229,144,  3, 76,138,194,173, 30,208,201,  3,208,  3, 76,  0,194
11160 DATA 76,206,194, 96,224,  0,240,  3, 76,232,194,173, 16,208, 41,254
11170 DATA141, 16,208,162,255, 76,232,194, 96,134,250,132,251,173,197,  0
11180 DATA201,  6,208, 13,206,  5
11190 DATA208,173,  5,208,201,106,176,  3,238,  5,208,234,234,234,201,  3
11200 DATA208, 13,238,  5,208,173,  5,208,201,221,144,  3,206,  5,208,234
11210 DATA234,234,173,141,  2,201,  1,240, 20,201,  2,208, 29,206,  3,208
11220 DATA173,  3,208,201,106,176,  3,238,  3,208, 76,105,195,238,  3,208
11230 DATA173,  3,208,201,221,144,  3,206,  3,208,166,250,164,251, 96,169
11240 DATA  0,133,252,169
11245 DATA 254
11250 DATA133,253,230,252,208,252,230,253,208,248, 96
11260 DATA  0,255
```

Combinatoria 64



El mes de septiembre para un estudiante principalmente supone dos cosas: el final de las vacaciones y la vuelta al cole o instituto. Al enfrentarse al nuevo curso, muchas cosas que se aprendieron el curso pasado, al final se han olvidado, aunque no el concepto sí el desarrollo.

Pues bien, el cuadernillo de este mes, y dado que una gran parte de los lectores de esta revista son estudiantes, le vamos a dedicar a sencillos programas, que de forma entretenida os ayudarán con ese repaso, que siempre resultará más ameno ayudado por el Commodore.

Este primer programa te permitirá resolver problemas de permutaciones, combinaciones y variaciones, con repetición o sin repetición. Fue realizado para la revista por Francisco Molina desde Barcelona.

Tras la breve presentación, aparece el primer menú con las siguientes opciones:

F1—Combinatoria con repetición.
F2—Combinatoria sin repetición.

Resaltamos que todo el programa está realizado a base de menús, lo cual facilita extremadamente el manejo del mismo, sin posibilidad de llegar a perderse y no saber por dónde continuar.

En un segundo menú se da a elegir entre permutación, variación o combinación, y para su selección únicamente se deberá pulsar el número que precede a cada una de ellas.

Una vez pulsada la opción que más interese, aparece en pantalla la fórmula correspondiente a esa opción, y luego la posición que ocuparían los parámetros introducidos en esa fórmula.

Como podréis observar su manejo es sencillo y su utilidad muy interesante. El mayor inconveniente quizás esté en la extensión del listado, en el que observaréis que está realizado sin utilizar controles de cursor, resultando por otra parte muy sencillo de comprender, con la posibilidad de adaptarlo a otros ordenadores.

La estructura seguida al realizar el programa está claramente explicada y señalada a través de instrucciones REMs en el listado. Esto te puede ayudar a modificarlo y hacer un programa que con poco más de la mitad de líneas realice la misma función.

```
10 REM***COMBINATORIA***
20 GOSUB2010
30 GOSUB1910
40 PRINTCHR$(147)
50 PRINTTAB(5)"COMBINATORIA SIN REPETICION"
60 PRINTTAB(4)"-------------------------------"
70 PRINTTAB(89)"1-..... PERMUTACION"
80 PRINTTAB(89)"2-..... VARIACION"
90 PRINTTAB(89)"3-..... COMBINACION"
100 GETA$:IFA$=""THEN100
110 A=VAL(A$)
120 ONAGOTO130,350,630
130 REM*PERMUTACIONES SIN*
140 PRINTCHR$(147)
150 PRINTTAB(6)"COMBINATORIA SIN REPETICION"
```



```
160 PRINTTAB(47)"P E R M U T A C I O N E S"
170 PRINTTAB(45)"-------------------------------"
180 PRINTTAB(87)"PARA LA FORMULA P  =  M!"
190 PRINTTAB(24)"M"
200 PRINTTAB(88)"INTRODUCIR PARAMETROS":P=1
210 PRINTTAB(91)"PARAMETRO  M=";:INPUTM
215 IFM>33THEN140
220 IFM<>INT(M)THENPRINTTAB(15)"ERROR":GOTO210
230 FORI=1TOM
240 P=P*I
250 NEXT
260 PRINTCHR$(147)
270 PRINTTAB(6)"COMBINATORIA SIN REPETICION"
280 PRINTTAB(47)"P E R M U T A C I O N E S"
290 PRINTTAB(45)"-------------------------------"
300 PRINTTAB(87)"PARA LA FORMULA P  =";M;"!"
310 PRINTTAB(23);M
320 PRINTTAB(90)"RESULTADO  P =";P
330 GOSUB2140
340 GOTO30
350 REM*VARIACION SIN*
360 PRINTCHR$(147)
370 PRINTTAB(6)"COMBINATORIA SIN REPETICION"
380 PRINTTAB(48)"V A R I A C I O N E S"
390 PRINTTAB(45)"-------------------------------"
400 PRINTTAB(92)"PARA LA FORMULA"
410 PRINTTAB(47)"N"
420 PRINTTAB(5)"V   =  M * (M-1)*(M-2)..."
430 PRINTTAB(7)"M"
440 PRINTTAB(89)"INTRODUCIR PARAMETROS";TAB(90);:V=1:C=0
450 INPUT"PARAMETRO  M=";M
455 PRINTTAB(90);
460 INPUT"PARAMETRO  N=";N
470 IFM<>INT(M)ORN<>INT(N)THENPRINTTAB(15)"ERROR":GOTO450
480 M1=M
490 V=V*M
500 M=M-1:C=C+1
510 IFC<NTHEN490
520 IFV<OTHENPRINTTAB(7)"ERROR":GOTO2140
525 PRINTCHR$(147)
530 PRINTTAB(6)"COMBINATORIA SIN REPETICION"
540 PRINTTAB(48)"V A R I A C I O N E S"
550 PRINTTAB(45)"-------------------------------"
560 PRINTTAB(91)"PARA LA FORMULA"
570 PRINTTAB(44);N
580 PRINTTAB(4)"V  = ";M1;"* (";M1"- 1 )";:IFN>2THENPRINT"*(";M1;"- 2 )...";
590 PRINTTAB(44);M1
600 PRINTTAB(90)"RESULTADO  V=";V
610 GOSUB2140
620 GOTO30
630 REM*COMBINACION SIN*
640 PRINTCHR$(147)
650 PRINTTAB(5)"COMBINATORIA SIN REPETICION"
660 PRINTTAB(46)"C O M B I N A C I O N E S "
670 PRINTTAB(44)"-------------------------------"
680 PRINTTAB(91)"PARA LA FORMULA "
690 PRINTTAB(90)"N";SPC(11);"M!"
700 PRINTTAB(8);"C   = ---------------"
710 PRINTTAB(10)"M    N! * (M - N)!
720 PRINTTAB(88)"INTRODUCIR PARAMETROS";TAB(90);
730 INPUT"PARAMETRO  M=";M
735 PRINTTAB(90);
740 INPUT"PARAMETRO  N=";N
```

```
750 IFM<>INT(M)ORN<>INT(N)THENPRINT"ERROR":GOTO730
755 IFM>33ORN>33THEN640
760 D=1:K=1:F=1
770 FORI=1TOM
780 D=D*I
790 NEXTI
800 FORI=1TON
810 K=K*I
820 NEXT
830 FORI=1TO(M-N)
840 F=F*I
850 NEXT
860 C=D/(K*F)
870 IFC<0ORC<>INT(C)THENPRINTTAB(15)"ERROR":GOTO2140
880 PRINTCHR$(147)
890 PRINTTAB(5)"COMBINATORIA SIN REPETICION"
900 PRINTTAB(46)"C O M B I N A C I O N E S "
910 PRINTTAB(44)"-----------------------------------"
920 PRINTTAB(91)"PARA LA FORMULA "
930 PRINTTAB(89);N;SPC(9);M;"!"
940 PRINTTAB(8)"C    = ----------------------"
950 PRINTTAB(9);M;SPC(3);N;"! * (";M;"-";N;"!)"
960 PRINTTAB(90)"RESULTADO  C=";C
970 GOSUB2140
980 GOTO30
990 REM*COMBINATORIA CON REPETICION*
1000 PRINTCHR$(147)
1010 PRINTTAB(5)"COMBINATORIA CON REPETICION"
1020 PRINTTAB(4)"-----------------------------------"
1030 PRINTTAB(89)"1-.....PERMUTACION"
1040 PRINTTAB(89)"2-.....VARIACION"
1050 PRINTTAB(89)"3-.....COMBINACION"
1060 GETA$:IFA$=""THEN1060
1070 A=VAL(A$):ONAGOTO1080,1600,1850
1080 REM*PERMUTACIONES CON*
1090 PRINTCHR$(147)
1100 PRINTTAB(6)"COMBINATORIA CON REPETICION"
1110 PRINTTAB(47)"P E R M U T A C I O N E S"
1120 PRINTTAB(45)"-----------------------------------"
1130 PRINTTAB(92)"PARA LA FORMULA"
1140 PRINTTAB(85)"A,B,C...";SPC(12);"M!"
1150 PRINTTAB(4)"P";SPC(8);"= --------------------"
1160 PRINTTAB(5)"M";SPC(11);"A! * B! * C! ...."
1170 PRINTTAB(89)"INTRODUCIR PARAMETROS";TAB(92);:P=1:GR=1:TG=1:SG=0
1180 INPUT"PARAMETRO M=";M
1190 IFM<>INT(M)THENPRINTTAB(15)"ERROR":GOTO1180
1200 FORI=1TOM
1210 P=P*I
1220 NEXTI
1225 PRINTTAB(9);
1230 INPUT"NUMERO DE GRUPOS";G
1240 FORI=1TOG
1250 PRINTTAB(9)"ELEMENTOS GRUPO";I;
1260 INPUTA1(I)
1270 NEXT
1280 FORI=1TOG
1290 SG=SG+A1(I)
1300 NEXT
1310 IFSG=MTHEN1380
1320 PRINTTAB(200);TAB(206);"■ERROR■,LA SUMA DE ELEMENTOS "
1330 PRINTTAB(50);"■A DE SER IGUAL A M ■'
1340 PRINTTAB(41)"PULSE UNA TECLA PARA INTRODUCIR NUEVOS ";
```

```
1350 PRINTTAB(54)"PARAMETROS"
1360 GETA$:IFA$=""THEN1360
1370 SG=0:GOTO1080
1380 FORI=1TOG
1390 FORK=1TOA1(I)
1400 GR=GR*K
1410 NEXTK
1420 A2(I)=GR
1430 GR=1
1440 NEXTI
1450 FORI=1TOG
1460 TG=TG*A2(I)
1470 NEXTI
1480 P=P/TG
1490 PRINTCHR$(147)
1500 PRINTTAB(6)"COMBINATORIA CON REPETICION"
1510 PRINTTAB(47)"P E R M U T A C I O N E S"
1520 PRINTTAB(45)"--------------------------------"
1530 PRINTTAB(92)"PARA LA FORMULA"
1540 PRINTTAB(85);A1(1);",";A1(2);"...";SPC(10);M;"! "
1550 PRINTTAB(4)"P";SPC(8)"=    --------------------"
1560 PRINTTAB(5);M;SPC(10);A1(1);"! *";A1(2);"! * ";A1(3);"! ...."
1570 PRINTTAB(91)"RESULTADO P=";P
1580 GOSUB2140
1590 GOTO30
1600 REM*VARIACIONES CON*
1610 PRINTCHR$(147)
1620 PRINTTAB(5)"COMBINATORIA CON REPETICION"
1630 PRINTTAB(48)"V A R I A C I O N E S"
1640 PRINTTAB(4)"------------------------------"
1650 PRINTTAB(90)"PARA LA FORMULA"
1660 PRINTTAB(96)"N";SPC(6);"N"
1670 PRINTTAB(14)"V̇    =  M"
1680 PRINTTAB(16)"M"
1690 PRINTTAB(48)"INTRODUCIR PARAMETROS";TAB(90);
1700 INPUT"PARAMETRO      M=";M
1705 PRINTTAB(90);
1710 INPUT"PARAMETRO      N=";N
1720 IFM<>INT(M)ORN<>INT(N)THENPRINT"ERROR":GOTO1700
1730 V=M↑N
1740 PRINTCHR$(147)
1750 PRINTTAB(5)"COMBINATORIA CON REPETICION"
1760 PRINTTAB(48)"V A R I A C I O N E S"
1770 PRINTTAB(44)"------------------------------"
1780 PRINTTAB(90)"PARA LA FORMULA"
1790 PRINTTAB(95);N;SPC(7);N
1800 PRINTTAB(13)"V";SPC(5);"=";SPC(4);M
1810 PRINTTAB(15);M
1820 PRINTTAB(49)"RESULTADO  V=";V
1830 GOSUB2140
1840 GOTO30
1850 REM*COMBINACIONES CON*
1860 PRINTCHR$(147)
1870 PRINTTAB(250)"LAS COMBINACIONES NO "
1880 PRINTTAB(85)"TIENEN REPETICION DE ELEMENTOS"
1890 GOSUB2140
1900 GOTO30
1905 REM*MENU BASE*
1910 PRINTCHR$(147)
1920 PRINTTAB(14)"COMBINATORIA"
1930 PRINTTAB(9)"----------------------"
1940 PRINTTAB(44)"TIENE LAS SIGUIENTES OPCIONES :"
1950 PRINTTAB(83)"F1-  COMBINATORIA CON REPETICION"
```

PROGRAMAS

```
1960 PRINTTAB(83)"F3-  COMBINATORIA SIN REPETICION"
1970 GETA$:IFA$=""THEN1970
1980 IFA$=CHR$(133)THEN990
1990 IFA$<>CHR$(134)THEN1970
2000 RETURN
2005 REM*PRESENTACION*
2010 PRINTCHR$(147)
2020 PRINTTAB(162)"■.─  .─.  ┌╲╱┐┌─.  ─┬─        .─  |"
2030 PRINTTAB(2)"■|   |  | |    || |    |             |   | |"
2040 PRINTTAB(2)"■|   |  | |    | ├─┴.  |             ├─. ├─┤"
2050 PRINTTAB(2)"■`─  `──' |    | |___' ─┴─           | |   | ,"
2060 PRINTTAB(2)"■                                     `──'   ┴"
2070 PRINTTAB(40)"■";SPC(9);" .───."
2080 PRINTTAB(9);"| .─ |"
2090 PRINTTAB(9);"||   |"
2100 PRINTTAB(9);"| `─ |■OPYRIGHT BY ■"
2110 PRINTTAB(9);" `───'"
2120 PRINTTAB(91);"FRANCISCO MOLINA"
2130 FORH=1TO6000:NEXT:RETURN
2140 REM*MENU OPCION FINAL*
2150 FORK=1TO4000:NEXT
2160 PRINTTAB(206)"DESEA MAS OPERACIONES?(S/N)"
2170 GETA$:IFA$=""THEN2170
2180 IFA$="S"THEN30
2190 IFA$<>"N"THEN2170
2200 PRINTCHR$(147):END
```

CODIGOS DE CONTROL PARA EL VIC-20 Y EL C-64

Colores del VIC-20 y del 64

Cómo se ve	Cómo se teclea	Efecto conseguido
	Ctrl+1	Negro
	Ctrl+2	Blanco
	Ctrl+3	Rojo
	Ctrl+4	Cian
	Ctrl+5	Púrpura
	Ctrl+6	Verde
	Ctrl+7	Azul
	Ctrl+8	Amarillo

Colores del 64 solamente

Cómo se ve	Cómo se teclea	Efecto conseguido
	Cbm+1	Naranja
	Cbm+2	Marrón
	Cbm+3	Rosa
	Cbm+4	Gris oscuro
	Cbm+5	Gris medio
	Cbm+6	Verde claro
	Cbm+7	Azul claro
	Cbm+8	Gris claro

Códigos de cursor y control

Cómo se ve	Cómo se teclea	Efecto conseguido
	Home	Cursor a casa
	Shift+home	Limpia pantalla
	Crsr	Cursor derecha
	Shift+crsr	Cursor izquierda
	Crsr	Cursor abajo
	Shift+crsr	Cursor arriba
	Ctrl+9	Carácter inverso
	Ctrl+0	Carácter normal
	Del	Borrar
	Shift+del	Insertar

Teclas de función

Cómo se ve	Cómo se teclea
	F1
	F2=Shift+F1
	F3
	F4=Shift+F3
	F5
	F6=Shift+F5
	F7
	F8=Shift+F7

commodore Magazine SERVICIO



Núm. 5 - 250 Ptas.
Programas, juegos y concurso/Londres: Quinta feria de Commodore/Basic, versión 4.75.



Núm. 6 - 250 Ptas.
El misterio del Basic/Lápices ópticos para todos/Concurso, juegos, aplicaciones.



Núm. 7 - 250 Ptas.
El ordenador virtuoso. MusiCalc. Programa monitor para el 64. Lápices ópticos. Ampliación de memoria para Vic-20.



Núm. 8 - 250 Ptas.
Joystick y Paddle para todos. Misterio del BASIC. EL LOGO. Cálculo financiero. Programas.



Núm. 9 - 250 Ptas.
Conversión de programas del Vic-20 al C-64. Móntale un paddle. Identifica tus errores. Software comentado.



Núm. 10 - 250 Ptas.
Koala Pad: La potencia de un paquete gráfico. Trucos. El FORTH. Software comentado. El LOGO.



Núm. 11 - 250 Ptas.
Music-64. Supervivencia (1.ª parte). Cómo guarda el diskette la información. Sintetizador-64. El Forth (1.ª parte).



Núm. 12 - 250 Ptas.
Commodore-16 por dentro y por fuera. Sprites: los alegres duendecillos (1.ª parte). Supervivencia (1.ª parte). El Forth (y 3.ª parte).



Núm. 13 - 250 Ptas.
Análisis: programas de ajedrez. Los Cazafantasmas, 64. Vic en el espacio. La impresora que dibuja. Interface paralelo.



Núm. 14 - 250 Ptas.
Sprites; cómo entenderse con los duendes. Pilot: un lenguaje de alto nivel. Guía de Software para C-64.



Núm. 15 - 250 Ptas.
Síntesis de voz: su ordenador tiene la palaba. Pilot: un lenguaje de alto nivel (2.ª parte). Guía de software para C-64 (2.ª parte).



Núm. 16 - 250 Ptas.
Análisis de simuladores: vuela con tu C-64. Contabilidad Cómo acelerar la ejecución de gráficos en BASIC. Submarino Commander. Pilot: (3.ª Parte).

DE EJEMPLARES ATRASADOS

Complete su colección de COMMODORE MAGAZINE

A continuación le resumimos el contenido de los ejemplares aparecidos hasta ahora.



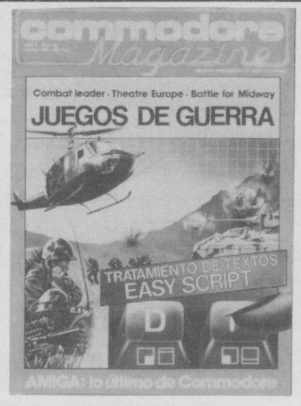
Núm. 17 - 250 Ptas.
Una lección de anatomía: los microordenadores por dentro. Bruce Lee: la furia oriental en el C-64. Quick Data Drive. Colossus Chess.



Núm. 18 - 250 Ptas.
Practicalc: todo el poder de una hoja electrónica. Pascal (1.ª parte). Programas: juegos y aplicaciones veraniegos. ¡Canasta!: dos ases del baloncesto para el C-64.



Núm. 19 - 250 Ptas.
Análisis de cuatro impresoras. Simulación: pequeños mundos en su ordenador. Pascal (2.ª parte). Entombed y The Staff of Karnath: aventuras gráficas y mucha acción.



Núm. 20 - 250 Ptas.
Juegos de Guerra: Combat leader, Theatre Europe, Battle for Midway. Tratamiento de textos Easy Script. Amiga: lo último de Commodore.



Núm. 21 - 250 Ptas.
Video-Digitizer: visión artificial para Commodore. Seikosha GP 700 VC: una impresora a todo color y con capacidades gráficas. Sprites multicolores. El nuevo C-128.



Núm. 22 - 300 Ptas.
Programas lightning: gráficos profesionales a tu alcance. Montaje: un interruptor programable para el C-64. Sprites múltiples. Cómo graba los datos el Datassette.



Núm. 23 - 300 Ptas.
Sinfonías en Chip: síntesis de sonido. Sprites en movimiento. Paisajes fractales en tu Commodore. Código máquina. Sight and sound: cuatro maestros de la música. Galería de Software.



Núm. 24 - 300 Ptas.
El ordenador en la enseñanza. ADA para Commodore. El C-128 en fotos. Los peques y el ordenador. Seikosha Sp-1000 VC. Investigación sobre el sonido.



Núm. 25 - 300 Ptas.
Los ports: conexiones al exterior. El BASIC del C-128. Matemáticas por ordenador. Software educativo. Melodías musicales. del Solfeo al Basic.



Núm. 26 - 300 Ptas.
Joysticks: ¿cómo son?, ¿cuáles son?, ¿cómo se programan?. Gráficos en código máquina. Simulador Spectrum. Proyecto Atenea.

Corte y envíe este cupón a: COMMODORE MAGAZINE
Bravo Murillo, 377 - Tel. 7337969 - 28020-MADRID

SERVICIO DE EJEMPLARES ATRASADOS

Ruego me envíen los siguientes ejemplares atrasados de COMMODORE MAGAZINE

El importe lo abonaré:
Contra reembolso □ Adjunto Cheque □ Con mi tarjeta de crédito □
American Express □ Visa □ Interbank □ Fecha de caducidad:

Número de mi tarjeta □□□□□□□□□□□□□□□□□□□□□

NOMBRE

DIRECCION

CIUDAD D.P.

PROVINCIA



Hablemos de ordenadores

o de cómo conseguir que no nos entienda nadie

No hay nadie que no se sienta sorprendido de no entender nada cuando escucha una conversación en el autobús que suena a castellano y, sin embargo, no parece serlo. —«Tengo un micro (microordenador) muy potente, tiene cepeeme (CP/M) concurrente, ciento veintiocho cas ram (128k RAM), interfeis serie y paralelo (INTERFACE CENTRONICS Y RS-232), además de un maus (ratón) y dos draibs (drives) de cinco y cuatro (5.25").»

Entre paréntesis podemos hallar las denominaciones más habituales de estas cosas a la que se refiere en su jerga el orador. Si somos, como normalmente suele ocurrir, unos aprendices del campo de la informática, todo este guirigay nos habrá sonado a chino, y podemos llegar a la conclusión de que el que así habla es un Gran Maestro en el Esotérico Mundo de la Informática, algo así como un Sumo Sacerdote. Pues no, es altamente probable que nos equivoquemos y que no sea más que un ilustre Engañado por la Alta Terminología de la impresionante Sonoridad.

Ante tamañas confusiones, la dirección de esta revista ha decidido encender un gran Faro en el Ténebre Mundo del Habla y concederé a los lectores la llama de la Buena y Castella Expresión. Para ello he elaborado el siguiente «DICCIONARIO ESPANGLIS-HESPAÑOL», compendio de los términos de uso más frecuente en inglés, su fea versión en espanglish y su versión más correcta en castellano.

No es completo este estudio, ya que hay muchas de estas palabras que no tienen traducción directa, y si la tienen suenan aún peor que el original.

Uno de los puntos cruciales en un ordenador es la memoria de que dispone. Pues bien, aquí es donde comenzamos a fracasar miserablemente en nuestro intento de hablar bien. La memoria se divide en dos bloques principales RAM y ROM (pronunciadas tal cual se escriben). La primera es la que el usuario tiene a su disposición para equivocarse, y la segunda es aquélla en la que se equivocó el fabricante (definición sacada de la experiencia propia y ajena: todos se quejan de los fallos de los ordenadores y nadie hace nada completamente a derechas).

En castellano, sus nombres son:

RAM = Memoria de Acceso Aleatorio (MAA).

ROM = Memoria de Sólo Lectura (MSL).

Como nadie nos entendería si hablásemos de la MAA o de la MSL, nos vemos obligados, a nuestro pesar, a recomendar que se hable de memoria *RAM* y *ROM*.

Otro punto a estudiar es las agrupaciones de memoria. En el punto más elemental tenemos el



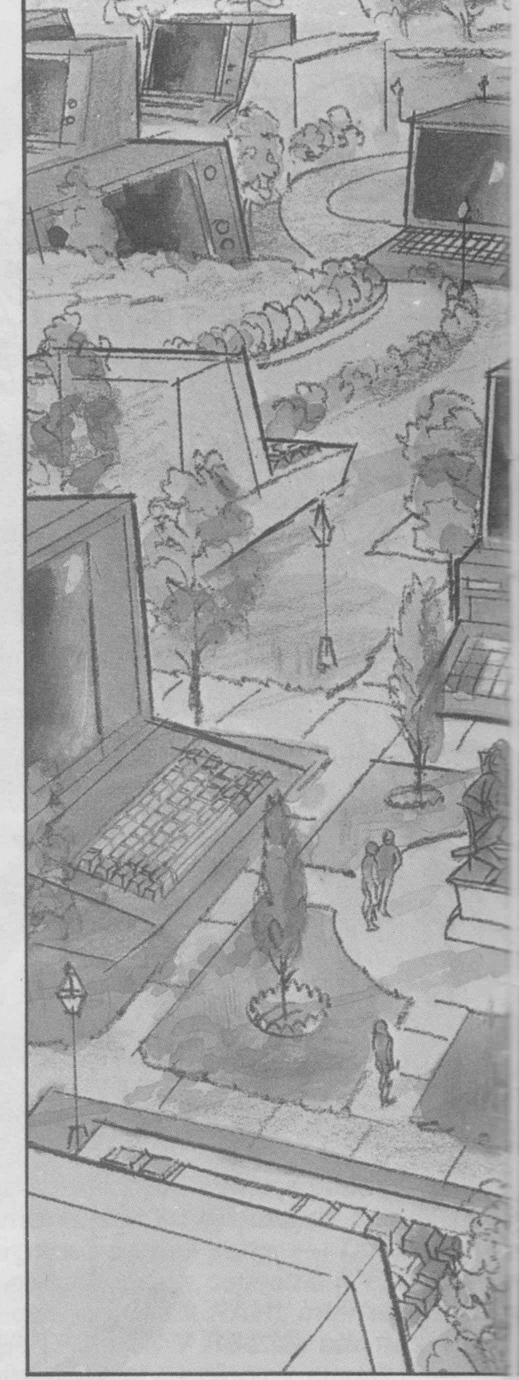
bit, que es un tonto que dice «sí» o «no» (nota: la definición de ordenador es: un millón de tontos diciendo SI y NO muy rápido), lo que clásicamente se dice «1» o «0».

Su nombre proviene de las palabras inglesas *Binary Digit* = Dígito Binario, lo que nos lleva a sugerir como nombre castellano DIO. Como siempre, nadie nos hará caso y le seguirá llamando BIT.

La siguiente unidad es el *NIBBLE*, palabra intraducible que corresponde 4 *bits*. Se pronuncia«nibel» y así habremos de llamarla, mal que nos pese. Siguiendo nuestra escalada, nos encontramos con el *«byte»*, 8 *bits*, pronunciado bait. Su traducción es MORDISCO, que todavía nadie ha aceptado como denominación española, a pesar de ser tan cómica como la inglesa (sólo que comprensible). En cambio, se emplea habitualmente una denominación tan anodina como OCTETO.

Otra unidad, menos usada, es la *«word»*, normalmente (no siempre) de dos bytes. La traducción directa, que es de hecho la que se usa, es PALABRA.

En términos mayores, se habla de *Kbytes*, o siempletemente K, que es un conjunto de 1.024 octetos. La denominación mejor sería KILOOCTETOS, pero, por ser muy larga, damos la alternativa de usar el término «K» (pronuncian-



do CAS). Cuando uno utiliza aparatos de mayor volumen, pueden hablar de los MEGAS y los GIGAS (1.024K y 1.024Gigas, respectivamente), que por tratarse de prefijos griegos, admitiremos como válidos.

Este apartado, como hemos podido observar, no ha sido un gran avance, ya que no hemos castellanizado casi ninguna palabra, pero os garantizamos que en apartados posteriores, daremos pasos de importancia.

LOS PERIFERICOS: COMO HABLAN LOS ORDENADORES

En un primer nivel, el ordenador habla con nosotros a través de una pantalla —televisor— monitor y de un teclado. Como se puede ver, hasta aquí no se presenta ningún problema de terminología, todo es buen castellano. Pero a veces nos hablarán del CRT, que es la denominación común a televisión y monitor, que en castellano se llama Tubo de Rayos Catódicos o, más abreviadamente, TRC. Otra posibilidad es el *LED-Display;* en castellano, la pantalla —LED, o bien el LCD— Display, es decir, la pantalla de cristal líquido.

Como alternativa al teclado tenemos el *«mouse»* (RATON), la bola gráfica el *«joystick»* (MANDO DE JUEGOS) y el tablero gráfico, todos ellos con usos muy específicos.

Cuando queremos enchufar una impresora a nuestro ordenador, nos esfrentamos a la alternativa que nos presenta el vendedor, nos enfrentamos a la alterna-o *RS-232?* Esto que nos dice el vendedor no son dos marcas de impresoras, son dos modos de hablar el ordenador con la impresora. Al primero, por comodidad, le llamaremos transmisión paralelo, y al segundo, transmisión serie. Estos nombres son generalizadores, ya que transmisión paralelo también es el sistema *IEEE-488* y paralelo el *RS-422*, pero como no son muy habituales los dejamos de lado. El sistema paralelo, lo que hace es enviar un bit por cable, por lo que envía un octeto cada vez, lo que es rápido pero caro al exigir 8 cables para la transmisión. En cambio, el sistema serie, envía los 8 *BITS* uno detrás de otro, por un solo cable, lo que lo hace más barato pero más lento. De todos modos, no se ha de elegir el sistema por economía, sino por el ordenador que se tenga. En el caso del Commodore el sistema que tiene implementado es CENTRONICS, es decir, paralelo.

Otros periféricos de interés son los discos, de los que hay varias categorías:

DISCOS FLEXIBLES: mal llamados *FLOPPY DISCS,* los hay en tamaños de 5,25 pulgadas y 12 pulgadas.

DISCOS CERAMICOS: en tamaños de 3, 3,25, 3,5, 3,9 pulgadas. El utilizado por el Commodore Amiga es el de 3.5 pulgadas, que es el tamaño que se está imponiendo en todo el mundo.

DISCOS DUROS: también llamados (mal) *HARD DISKS* o *WINCHESTERS*, sirven para almacenar mucho y muy rápido, y sólo tienen utilidad real para grandes sistemas, oficinas y programadores profesionales.

En general, a los discos flexibles y cerámicos se les llama *DISKETTES*, lo que es vulgar y malsonantes hasta en inglés. Como nombre corto para ellos, proponemos el que se usa desde siempre entre profesionales, DISCOS, ya que a los discos duros se les da el nombre completo. No hay confusión posible.

Existen muchos otros periféricos pero su uso es menos habitual y, por tanto, aparecen mucho menos en las conversaciones.

Dentro de este apartado cabe tratar también de la terminología técnica que se utiliza (por suerte, pocas veces) para describir la interacción entre el periférico y el microprocesador, en sus niveles más íntimos.

Por ejemplo, se habla de *«handshake»* («jancheik»). Esto significa APRETON DE MANOS, y quiere decir el intercambio de señales o controles que realizan periférico y procesador para poder llegar a un traspaso de datos eficiente. Viene a ser una especie de:

—«Oye, que te envío un bloque de datos»

—«Un momento, que ahora no te puedo hacer caso»

—«Vale, espero»

...

—«Chico, que ya puedo recibir»



—«Bueno, pues cuando suene la tercera señal, te los envío»

...

Esto, que es imprescindible (hablando se entiende la gente), el *«handshake»,* en castellano lo podemos denominar como PROTOCOLO, por su gran parecido con las ceremonias diplomáticas y cortesanas.

EL BASIC: TRADUCIR Y NO ADAPTAR

Hay muchos bárbaros sueltos entre la gran masa de aficionados a los ordenadores que son capaces de destrozar el castellano hasta su misma fuente. Allá donde es muy sencillo decir, por ejemplo, IMPRIMIR, los muy Atilas dirán *PRINTER*. Este es uno sólo de los múltiples ejemplos que podemos hallar. Como casi todos ellos se producen en usuarios de BASIC y son fruto de la adaptación de instrucciones inglesas a una nomenclatura con una cierta sonoridad castellana.

Aparte de la burrada ya mencionada, cabe denotar las siguientes, todas ellas reales (soy testigo):

IMPUTEAR por PREGUNTAR o PEDIR

FORMATEAR por DAR FORMATO

PLOPEAR por PINTAR (UN PUNTO)

GOSUBEAR por LLAMAR A UNA SUBRUTINA

ERASEAR por BORRAR

etcétera...

Mi recomendación es que, en caso de duda o de que sea muy larga la traducción en castellano, se utilice el original inglés sin modificación, pues por lo menos se tienen garantías de que es una palabra válida y existente en algún idioma.

UN VIAJE AL INTERIOR (I): EL SOPORTE LOGICO

Supongo que la primera duda que aparece ante este título es «¿Qué es eso de Soporte Lógico?» Pues bien, el soporte lógico es lo que vulgarmente se llama «software» (pronunciado «sofgüer»). El título por sí es ya una declaración de principios sobre lo que queremos conseguir con este artículo: hablar español.

Dentro del soporte lógico, cabe destacar tres estructuras a analizar:

—El sistema operativo.
—Los lenguajes.
—Los programas.

El sistema operativo es aquella parte que viene con el ordenador y que sirve para su comunicación con el mundo exterior a través de los periféricos.

El sistema operativo de un ordenador incluye una serie de rutinas para conseguir realizar tareas como grabar en disco o cinta, recuperar datos de ellos, dibujar en pantalla o impresora, hacer ruidos, etc.

Los lenguajes son lo que nos permiten hacer con el ordenador lo que nos apetezca (siempre que sepamos cómo). Estos lenguajes tienen tres variantes importantes, a saber: el código de máquina, los intérpretes y los compiladores.

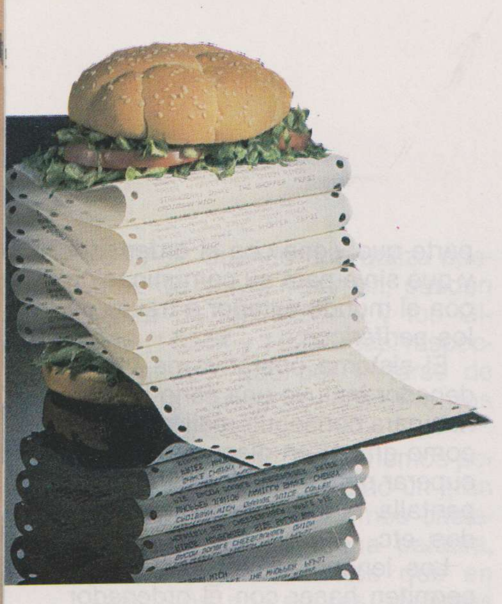
El código de máquina es un conjunto de unos y ceros que sirven para que el microprocesador ejecute una serie de acciones sencillas (saltar, sumar, complementar o transferir datos) y que es, de hecho, el único lenguaje que entiende.

Un intérprete lo que hace es leer un programa escrito en un lenguaje distinto del código de máquina, instrucción a instrucción y efectuar una serie de operaciones en código de máquina (u otro lenguaje) para realizar la acción que se espera de él. Tal es el caso del intérprete de BASIC que lleva incorporado un ordenador Commodore.

Un compilador lo que hace es leer completo un programa escrito en algún lenguaje y traducirlo como conjunto a una serie de operaciones en código máquina. No olvidemos nunca que los ensambladores (traductores de lenguaje ensamblador a código de máquina) son los más sencillos de los copiladores.

Para aclarar las diferencias entre intérprete y compilador, supongamos el caso de una conferencia que se diese a gente de muy diversos países. Para hacerles comprender a todos lo que quiere decir, se optaría por una de dos alternativas:

a) Darles la conferencia ya tra-



ducida, con lo que sólo habrían de preocuparse de leerla. Esto es un compilador.

b) Poner un traductor simultáneo. Esto es lo que hace un intérprete, traducir simultáneamente.

Como última diferencia, destacaremos que un programa suele ejecutarse más lentamente al interpretarse que al compilarse.

Los programas se dividen en Aplicaciones y Juegos (principalmente). Las aplicaciones son cosas como procesadores de textos, bases de datos, hojas de cálculo (también llamados estallidos u hojas electrónicas)... Los juegos los conocen todos, por lo que no exigen más explicación.

VIAJE AL INTERIOR (II): EL SOPORTE FISICO

Del mismo modo que antes hablábamos del soporte lógico, el soporte físico es el *«Hardware»* (pronunciación: «jargüer»), y que consta de todo lo que tiene en realidad tangible dentro del ordenador.

Como elemento fundamental aparece el microprocesador, un bicho habitualmente negro grisáceo con muchas patas (en el Commodore se llama 6510 y tiene 40 patas). Este es el corazón del aparato, el que ordena y manda, siempre bajo tus órdenes.

Junto a él se hallan muchos otros bichos del mismo color. (Entre ellos tenemos el chip (traducido al español sería «pastilla»).

SID 6581 (Sound Interface Device = Dispositivo Interfaz de Sonido), también está el CIA 6526 (Complex Interface Adapter = Adaptador Inferfaz Complejo), otro es el VIC 6569 (Vídeo Interface Control = Controlador Interfaz de Vídeo) y algunos dispositivos lógicos. Todos ellos sirven para que el microprocesador pueda hacer algo útil.

El microprocesador se comunica con los arriba mencionados a través de una cosa llamadas «buses», y cuya más correcta traducción al castellano aparecida hasta ahora es, precisamente, «buses».

Estos buses con conjuntos de hilos por los que viajan unos y ceros (por eso se les llama buses, porque por ellos viajan cosas) y que cumplen muy diversas funciones, de las que depende el nombre que se le dé al bus por el que van.

En primer lugar, tenemos el Bus de Datos, que sirve para transferir datos entre el procesador y la memoria y periféricos. Para controlar este flujo, tanto su dirección como su significado, se utiliza el Bus Control. Finalmente, para controlar su destino, se utiliza el Bus de Direcciones.

Valga esta corta perorata para introducir a la vez la nomenclatura castellana y los fundamentos de operación del interior de un ordenador.

DESPEDIDA

No podemos negar que se nos han quedado algunas palabras de uso más o menos frecuente por traducir, pero son, en general, vocablos que no utilizamos normalmente en nuestra charla habitual. Por ello, para dichas palabras, remitimos a un diccionario inglés-castellano que si no nos dicen su traducción plenamente correcta, al menos nos da buenas ideas sobre lo que significa y cómo podríamos decirlo.



Como última recomendación, recordar que estamos en España y ello hace recomendable hablar en español. Pero recordad siempre: SI QUEREIS DESPISTAR O ASOMBRAR A VUESTROS AMIGOS, UTILIZAD TERMINOLOGIA INGLESA, Y SI ES POSIBLE, PRONUNCIADLA MAL. Esto último es a lo que se han dedicado durante el último decenio aquellos que han querido dársela de «enteraos».



Colisión entre sprites

P: El motivo de esta carta es para preguntarles sobre una duda que tengo con la detección de colisión entre sprites. Resulta que, por ejemplo, en el siguiente programa:

```
10 PRINTPEEK (V+30)
20 IFPEEK (V+30)= 2 THEN
PRINT "M"
30 IFPEEK (V+30)= 3 THEN
PRINT "A"
```

Ejecutando éste y apareciendo en pantalla un tres, que es la función de la línea 10, el programa al llegar a la línea 30 no imprime A. Me gustaría saber a qué es debido esto, y si puede ser un fallo en la estructura del programa, ya que en la mayoría de los casos esto se produce en programas más bien largos, o si hay alguna instrucción que no deje pasar el programa al imprimir un tres en pantalla.

Antonio Núñez
Barcelona

R: Una de las características particulares del chip VIC-II, en lo referente a su capacidad para detectar colisiones, es que la dirección de memoria 53278 ($D01E), es decir, PEEK (V+30) siendo V=53248, tiene la particularidad de que al ser leída se borra automáticamente, es decir, si lees dos veces la misma colisión, la segunda vez ya no es detectada por el ordenador. La solución a esto puede ser, guardar el contenido del registro al colisionar en una variable hasta que hayas realizado todas las operaciones que se deriven de una colisión.

En tu caso particular y por el programa ejemplo que nos envías, al leer en la instrucción 10 el valor del registro V+30, éste toma inmediatamente el valor 0, lo que te impide detectar las colisiones en las siguientes instrucciones.

La forma lógica de hacerlo, sería:

```
10 Z= PEEK (V+30): PRINT Z
20 IF Z= 3 THEN PRINT "COLISION"
```

Con el valor 3 detectarías la colisión entre el Sprite 0 y el Sprite 1, el valor 20 detectaría la colisión entre el Sprite 2 y el Sprite 4 y así puedes detectar la colisión deseada.

1	2	4	8	16	32	64	128

Un acierto

P: Se trata de una colaboración pensada para aquellos usuarios de Commodore 64 que utilicen una impresora STAR SG 10 con el interface SECUS, y en relación con el EASY SCRIPT.

Consiste en poner la impresora en modo IBM (switch 2-2 off) , y 1-7 y 1-8 off, aunque esto no es muy importante. Al preguntar el programa tipo de impresora, dar la opción 1 y en el tipo de interface, digitar "s" (Serial). De esta forma tendremos todas las vocales acentuadas, así como caracteres especiales que no podríamos tener en modo STAR. Los códigos coinciden con los que da el manual de la impresora para este modo (el IBM).

Tenéis que dar la dirección secundaria 25 donde se definen los caracteres especiales (sa25). Ej.:

*sa25
*1=160 para la á
*2=248 para la º
*3=161 para la í
*4=162 para la ó
*5=163 para la ú
*6=165 para la ñ
*8=166 para la ª

Y así podríamos continuar con todas las letras y signos, tales como ¡, ¿, subrayados, etc...

También para el retroceso de un espacio, podéis pulsar F1 + flecha a la izquierda (←) y para dar las secuencias de escape, pulsar F1 + E.

Fernando Jiménez
Madrid

R: Lo comprobamos, nos gustó y aquí está publicado.

Codigo Máquina

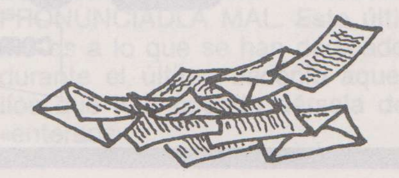
P: Os escribo para que me resolváis algunas dudas que tengo sobre el Código Máquina.

1.º ¿Hay alguna instrucción o instrucciones en código máquina que actúen como las instrucciones del Basic (DATA/READ). En caso negativo, ¿cómo puedo yo almacenar una GRAN cantidad de datos indispensables para la perfecta ejecución de un programa?

2.º En el libro de Mark Greenshields, «Commodore 64 Código Máquina avanzado» se citan al final del libro (concretamente en la sección tercera) las rutinas de la ROM. Mi pregunta es: ¿cómo puedo yo utilizarlas? y ¿qué posible aplicación tendrían en un programa?

3.º ¿Se podría hacer un programa en que una primera parte del mismo estuviese escrita en Basic y la segunda parte (por de-







cirlo de alguna manera) en código máquina? ¿Se pueden unir instrucciones de Basic y código máquina en una misma línea? por ejemplo:

PRINT LDA $OFFF

Ya sé que son muchas preguntas pero agradecería que me las respondiéseis lo antes posible.

Miguel López
Madrid

R: En primer lugar, y respondiendo a tu primera pregunta, no existe ningún equivalente a READ/DATA en Código Máquina y creo que no deberías intentar aprender este lenguaje comparándolo con el Basic, pues es totalmente distinto. En código máquina todo se almacena en direcciones de memoria que luego puedes extraer, simulando el READ/DATA, pero tiene más complicaciones que debes de tener en cuenta y que en este apartado resulta difícil de explicar.

En segundo lugar, las subrutinas que tiene la ROM de tu ordenador son las que te permite ejecutar programas en BASIC. Si conoces las posiciones de estas subrutinas y sus parámetros de entrada y salida puedes utilizarlas para simplificar la programación en código máquina y conseguir que tus programas sean más cortos. A estas subrutinas se acceden mediante el cómodo JSR y la dirección de memoria donde está ubicada. Estas subrutinas puedes encontrarlas especificadas en cualquier manual avanzado para el Commodore.

En respuesta a tu última pregunta, te podemos decir que no se pueden unir dos instrucciones de distintos lenguajes. La forma más sencilla de hacer una llamada desde un programa BASIC a otro en código máquina es mediante la instrucción SYS y la dirección de memoria donde empieza el programa en código máquina, este programa debe finalizar con la instrucción RTS.

Otras formas pueden ser, por ejemplo, utilizar la función USR, cambiar uno de los vectores de entrada y salida en RAM, cambiar uno de los vectores de interrupción en RAM, cambiar la rutina CHRGET.



Pa
es
FISHER

arámetros stadísticos

LA estadística es una de las ramas más útiles de las matemáticas a la hora de determinar ciertas cuestiones de interés práctico. Puede interesarnos, por ejemplo, tener una idea de qué valor posee por término medio, una determinada característica de un conjunto de elementos, o predecir un valor futuro con ciertas garantías de no cometer un gran error en nuestra estimación.

Supongamos que en un determinado curso de bachillerato hay 100 alumnos y que las alturas de dichos alumnos son las siguientes (que hemos ordenado en sentido creciente):

1.51	1.51	1.51	1.52	1.52	1.53
1.53	1.53	1.54	1.54	1.55	1.55
1.56	1.56	1.56	1.57	1.57	1.57
1.58	1.58	1.59	1.59	1.59	1.60
1.60	1.61	1.61	1.61	1.62	1.62
1.63	1.63	1.63	1.64	1.64	1.65
1.65	1.65	1.66	1.66	1.66	1.67
1.67	1.67	1.68	1.68	1.69	1.69
1.69	1.70	1.70	1.71	1.71	1.71
1.72	1.72	1.73	1.73	1.73	1.74
1.74	1.74	1.74	1.74	1.75	1.75
1.75	1.76	1.76	1.77	1.77	1.78
1.78	1.79	1.79	1.79	1.80	1.80
1.81	1.81	1.81	1.82	1.82	1.83
1.83	1.83	1.84	1.84	1.85	1.86
1.87	1.87	1.87	1.88	1.88	1.89
1.89	1.89	1.90	1.90		

En general no nos interesará saber cuál es la altura de cada uno de los alumnos, sino que nos bastará con algunos datos que reflejen cuáles son las características del conjunto. Por ejemplo, si decimos que estos 100 alumnos tienen una altura mínima de 1.51 m. y máxima de 1.90 m., que en general la altura de un alumno oscilará alrededor de los 1.70 m., con cierta probabilidad de que sea de 1.70 m. ± 0.11 m.; que la altura más frecuente entre los 100 alumnos es de 1.74 m., y que hay 50 alumnos que miden 1.70 m. o menos y 50 que miden 1.70 m. o más, tendremos una idea bastante aproximada del conjunto. Por eso, si tenemos una población compuesta por N individuos (los 100 alumnos) y queremos tener una idea bastante precisa de una característica (la altura), podemos tenerla calculando los siguientes parámetros, llamados **estadísticos:**

RECORRIDO (R).— Es la diferencia entre el valor más alto y el más bajo de la característica que se está estudiando (x).

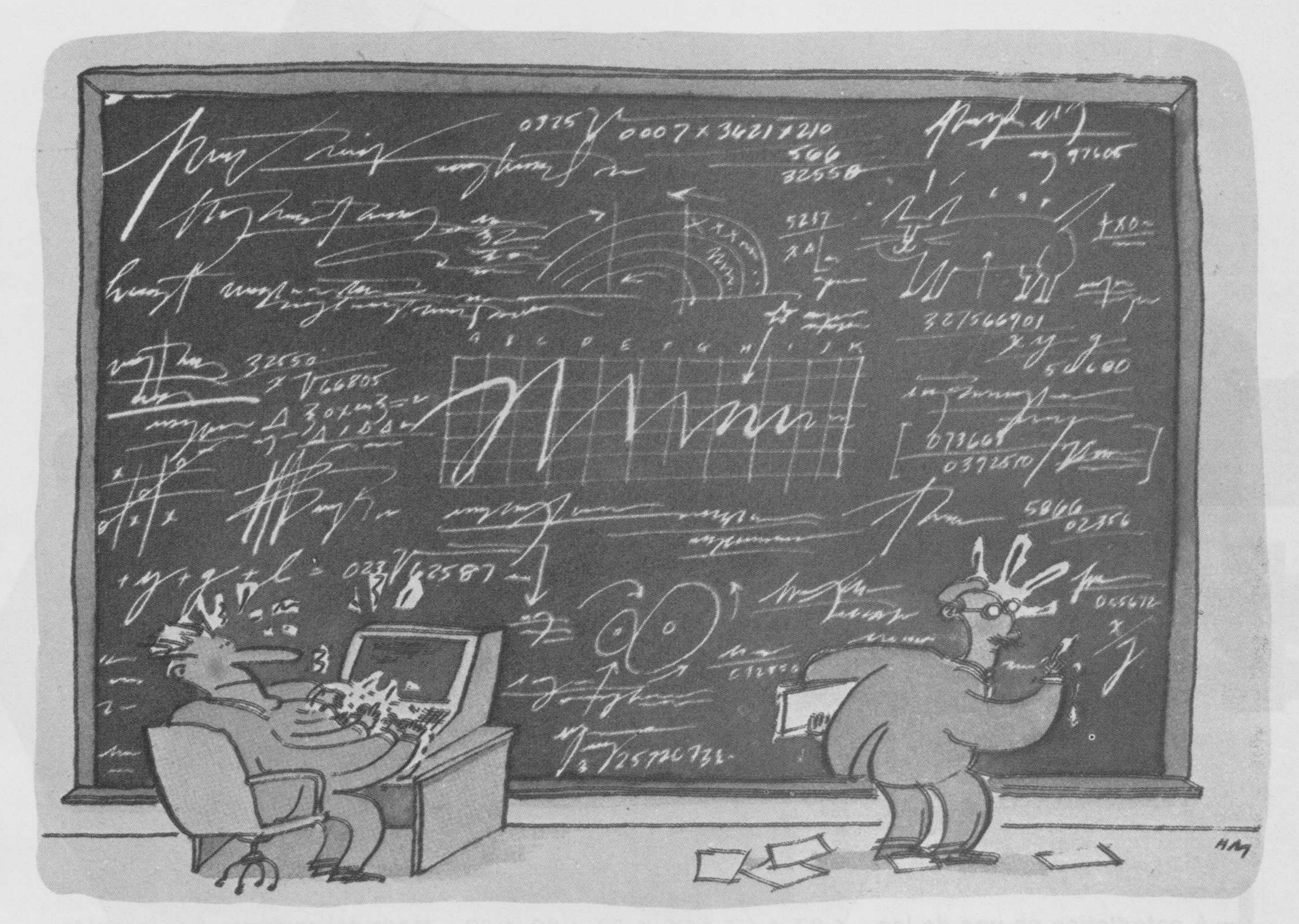
$$R = X_{máx} - X_{mín.}$$

En nuestro ejemplo es:

$$R = 1.90 - 1.51 = 0.39$$

El recorrido nos da una idea de la extensión del intervalo en que se encuentra confinada la característica estudiada en la muestra (es decir, en el conjunto de datos observados).

MEDIA ($\bar{x}$).— Es el cociente entre la suma de todos los datos y el número total de ellos que comprende la muestra. Esta definición de media se corresponde bastante fielmente con la



idea intuitiva de valor medio de un conjunto de datos.

$$\overline{x} = \frac{x_1 + x_2 + ... + x_n}{N} =$$

$$= \frac{\sum_{i=1}^{N} x_i}{N}$$

La altura media entre los 100 alumnos es, por tanto:

$$\overline{x} = 1.6996$$

DESVIACION TIPICA (σ).—Es la raíz cuadrada de la suma de los cuadrados de las diferencias entre cada dato y la media, dividida por el número total de datos:

$$\sigma = \sqrt{\frac{(x_1 - x)^2 + (x_2 - x)^2 + ... + (x_n - \overline{x})^2}{N}} =$$

$$= \sqrt{\frac{\sum_{i=1}^{N} (x_i - x)^2}{N}}$$

La desviación típica da una idea de entre qué valores de x puede esperarse que esté un caso determinado. Así si por algún procedimiento se obtiene que, para unas condiciones determinadas, x es igual a 3, y es igual a 0.5, se puede afirmar con cierta garantía de acierto que x está entre 3 — 0.5 = 2.5 y 3 + 0.5 = = 3.5. Sin embargo hay una pequeña probabilidad de que x esté fuera del intervalo (2.5-3.5). En nuestro caso:

$$\sigma = 0.112338$$

La **VARIANZA** (σ) es un estadígrafo muy importante, utilizado sobre todo en muestreo. Su valor es el cuadrado de la desviación típica:

$$\sigma^2 = \frac{\sum_{i=1}^{N} (x_i - \overline{x})^2}{N}$$

MODA (m).—Es el valor de x más frecuente. En nuestro ejemplo, la altura que más se repite es 1.74. Por tanto 1.74 será la moda.

MEDIANA (M).—Es el valor de x tal que el 50% de los casos sean inferiores a él y el 50% superiores.

Observando nuestra tabla de alturas encontramos que el 50% de los datos son inferiores (o iguales) a 1.70, y el 50% son superiores (o iguales), por tanto:

$$M = 1.70$$

En definitiva tenemos (en nuestro ejemplo) un conjunto de 100 alumnos con los siguientes valores estadísticos:

Tamaño de la muestra: 100
Recorrido: 0.39 m.
Media: 1.6996 m.
Desviación típica: 0.112338 m.
Varianza: 0.01262 m^2
Moda: 1.74 (frec. = 5)
Mediana: 1.70 m.

El siguiente programa BASIC nos permite, previa introducción de los correspondientes datos, calcular los estadísticos de que hemos hablado en esta ocasión.

Juan Antonio Feberero

```
10 REM PARAMETROS ESTADISTICOS
20 REM ************************
40 PRINT"[CLR]"
50 INPUT "NUMERO DE DATOS";N
60 PRINT
70 DIM X(N)
80 FOR I=1 TO N
90 PRINT I;
100 INPUT X(I)
110 NEXT I
120 XI=X(1):XS=X(1)
130 FOR I=2 TO N
140 IF X(I)<XI THEN XI=X(I)
150 IF X(I)>XS THEN XS=X(I)
160 NEXT I
170 PRINT"[CLR]"
180 PRINT "X INFERIOR..:";XI
190 PRINT "X SUPERIOR..:";XS
200 PRINT "RECORRIDO...:";XS-XI
210 XM=0:DT=0:MO=0
220 FOR I=1 TO N
230 XM=XM+X(I)
240 NEXT I
250 XM=XM/N
260 PRINT
270 PRINT"MEDIA.......:";XM
280 FOR I=1 TO N
290 DT=DT+(X(I)-XM)↑2
300 NEXT I
310 DT=DT/N
320 PRINT
330 PRINT "DESV. TIPICA:";DT↑.5
340 PRINT
350 PRINT "VARIANZA....:";DT
360 FOR I=1 TO N
370 MA=0
380 FOR J=1 TO N
390 IF X(J)=X(I) THEN MA=MA+1
400 NEXT J
410 IF MA>MO THEN MO=MA:IM=1
420 NEXT I
430 PRINT
440 PRINT "MODA........:";X(IM)
450 PRINT "FRECUENCIA..:";MO
460 ME=N/2
470 IF INT(ME)-ME<0 THEN ME=X(ME+1):GOTO 490
480 ME=(X(ME)+X(ME+1))/2
490 PRINT
500 PRINT "MEDIANA.....:";ME
510 END
```

Compatibles



El término «compatible» es muy utilizado en el mundo del PC. Este artículo introduce al lector en las particularidades y detalles que están detrás de él. No existe prácticamente ninguna casa de microordenadores que no ofrezca en la actualidad un «compatible» IBM.

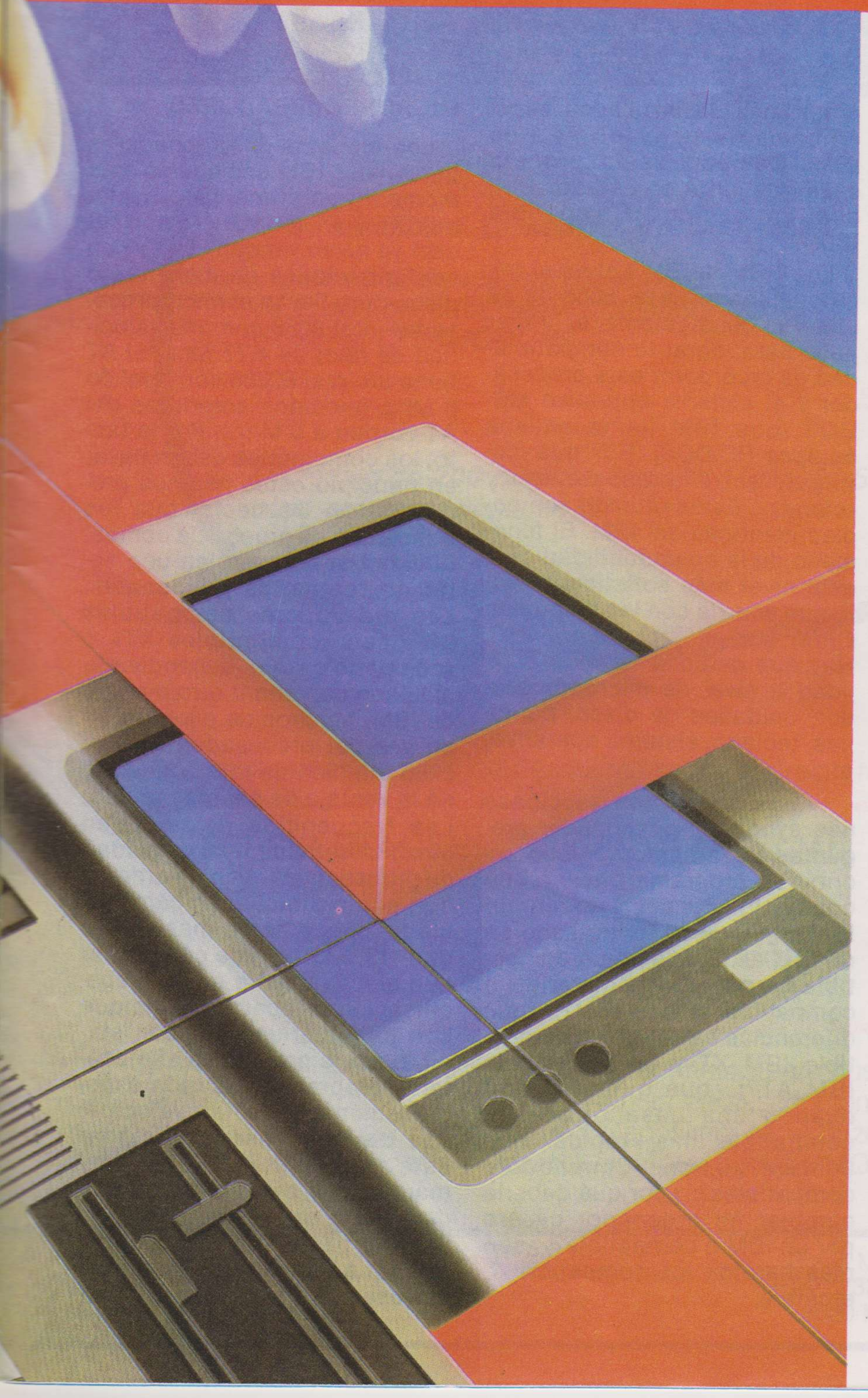
Los microordenadores han evolucionado constantemente desde que se lanzaron al mercado por primera vez los Apple y Commodore. Los primeros modelos tenían una capacidad limitada de memoria y cada uno ofrecía su propia versión del Basic y del Sistema Operativo. Poco a poco fueron apareciendo otras marcas y modelos. Los sistemas operativos se fueron perfeccionando y se incorporaron los discos de diversas capacidades y formatos. La difusión a gran escala de estos micros favoreció la creación de casas de software que ofrecían programas de hojas de cálculo, bases de datos y programas de contabilidad y gestión un tanto rudimentarios, pero que empezaban a tener una gran demanda. Por otra parte, los fabricantes de hardware necesitaban ofrecer al cliente potencial algo que pudiera interesarles e influyera al momento de hacer la decisión de compra.

ESTANDAR CP/M

El Basic de Microsoft y el Sistema Operativo CP/M de Digital Research conjuntamente con el microprocesador Z80 de Zilog habían demostrado ser elementos indispensables en cualquier micro. Por lo tanto fueron adoptados como estándar por muchos fabricantes. Hace ya varios años no era raro leer en un folleto de publicidad términos como: «Incluye CP/M» o «Incluye el Basic Microsoft», «microprocesador Z80», etc. Empezaba así de alguna forma lo que podría llamarse la «compatibilidad» en los micros. El argumento es bastante sencillo en principio. Si tengo un equipo que trabaja con el mismo sistema operativo y el mismo Basic que los que emplea el ordenador de mi amigo Pepe, quiere decir que un programa escrito para mi ordenador puede correr en el suyo.

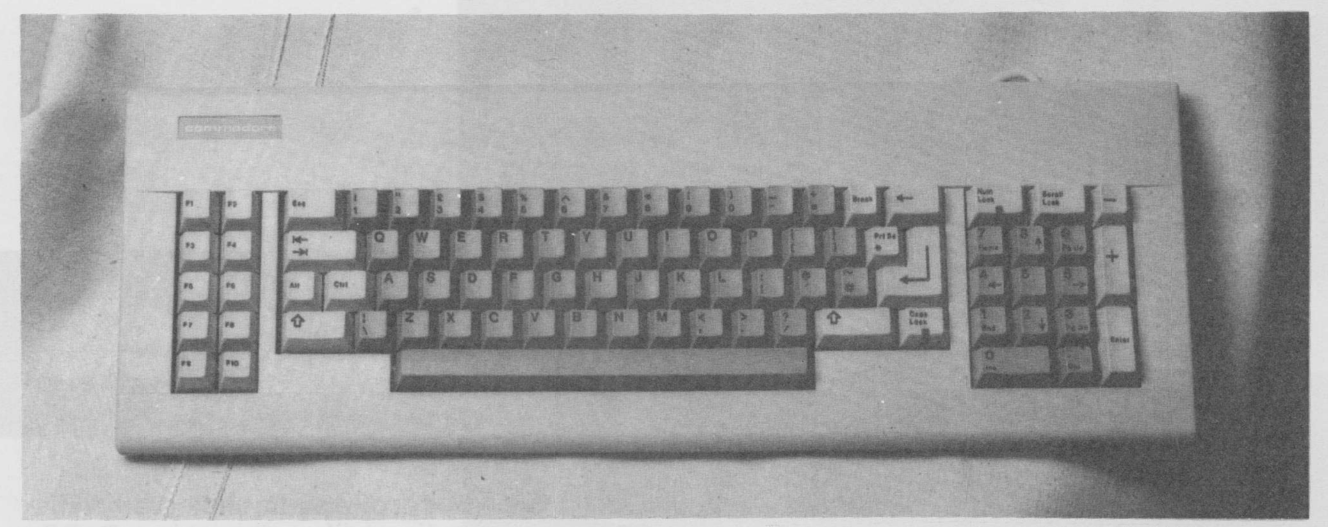
Sin embargo, no todo era tan simple, además del mismo Basic y del mismo Sistema Operativo, también tenían que ser iguales las pantallas, el teclado y el formato de los discos. De todas formas, un programa escrito para una determinada máquina correría en otra, siempre y cuando se volviera a escribir, salvo que esa máquina pudiera leer mis discos. Poco a poco se fueron ofreciendo unidades de disco con similares características para diversas marcas de micros, y algo parecido ocurría con los teclados. Las pantallas,

atibles Compatibles



por otra parte, tenían características diferentes, más o menos colores, 40, 64 u 80 columnas, juegos de gráficos diferentes, etc., hacían muchas veces incompatibles los programas al correrlos en diferentes máquinas. Una solución para este último problema era emplear sólo las funciones estándar del Basic soportadas por todas las máquinas y dejar de aprovechar las mejoras que cada fabricante incorporara en sus modelos. A pesar de todo había y sigue habiendo marcas como Apple, Commodore, Amstrad y otras que ofrecen sus propias versiones de Sistema Operativo o del Basic a nivel de ordenadores caseros. Hay que señalar que todos estos micros son de 8 bits y básicamente de 64 Kbytes direccionables directamente.

Podemos decir que en ése entonces había un estándar (unas normas que se consideran como referencia a seguir) determinado por el CP/M y el Basic Microsoft. Cualquier marca que quisiera competir con éxito en el mercado debería ser capaz de poder ofrecer estos estándares (salvo Commodore y Apple que mantenían sus propias versiones). Un factor determinante en esta adopción era la oferta de gran cantidad de software que corría bajo CP/M. Con el tiempo surge otro estándar en ordenadores caseros llamado el MSX. Se desarrollaron diversidad de marcas que ofrecían la particularidad de estar dotadas con las mismas unidades de disco, teclados, sistema operativo Basic, etc. Era, y es, una forma más perfeccionada de «compa-



tibilidad», es decir, equipos de diferentes marcas pero que son «iguales» de alguna forma.

Cuando IBM lanza al mercado su Ordenador Personal se crea una gran expectativa. Nunca antes había estado IBM en este mercado. Su nombre era ampliamente conocido en el mundo de la informática de las grandes empresas. Cuando IBM empieza su andadura en los micros, los 8 bits y las escasas 64 K de memoria que ofrecían los micros de entonces no eran suficientes para satisfacer las necesidades de mejores y más potentes programas. Por esta razón IBM da un salto e introduce un micro de 16 bits con capacidad de direccionar directamente 640 K bytes en la memoria central. Adopta el sistema operativo MS-DOS de Microsoft y el Basic también de esta casa.

ESTANDAR MS-DOS

Los modelos IBM PC e IBM XT (con disco duro) vendidos masivamente en todo el mundo crean un fenómeno interesante. Cuanto mayor es el número de usuarios más casas de software desarrollan programas para el ordenador IBM, y al mismo tiempo cuantos más programas hay escritos para IBM más compradores se inclinan por tener un IBM. De hecho esta mutua potenciación de ventas da lugar al reconocimiento de un nuevo estándar: el MS-DOS. Todos los fabricantes se dan cuenta enseguida de que tienen que ofrecer al cliente potencial una máquina capaz de correr los miles de programas desarrollados específicamente para los PCs de IBM. Compaq es la primera casa que lanza en Estados Unidos un equipo «Compatible IBM».

Los PCs de IBM tienen el microprocesador Intel 8088 de 16 bits. Por consiguiente, el primer paso para lograr la compatibilidad es incorporar este procesador y el sistema operativo MS-DOS (que IBM llama en sus equipos PC-DOS). Hay que distinguir entre «compatibles» y «clones». Estos últimos son copias idénticas del IBM. El fenómeno de la compatibilidad es a este nivel más sofisticado que lo que ocurría con las máquinas CP/M con el procesador Z80. Hay que estudiar la compatibilidad a nivel de microprocesador, unidades de disco, pantallas, teclado, sistema operativo, etc.

Las siguientes secciones nos ayudarán a responder preguntas tales como: ¿Puedo escribir un programa Basic en un compatible y correrlo en un IBM?; ¿un disco formateado en un IBM es legible en un compatible?; ¿por qué compraría un compatible si hay IBM?; ¿qué diferencias hay entre un compatible IBM XT y un compatible IBM AT?; ¿qué quieren decir cuando una casa ofrece un compatible más «rápido» que un IBM?; ¿tiene Commodore compatibles?; ¿por qué debería cambiar mi ordenador casero por un IBM o compatible?, ¿son más baratos los compatibles?, etcétera.

MICROPROCESADORES

Los modelos de ordenadores personales IBM son dos en este momento en el mercado nacional: XT y AT (el PC y el portátil casi ya no se venden). El XT con configuraciones de diskettes o disco duro llevan el microprocesador Intel 8088 con una velocidad de reloj de 4,77 Mhz. El AT tiene un Intel 80286 con reloj de 8 Mhz. (modelos anteriores del AT corrían a 6 Mhz.). Por lo tanto, los compatibles deben llevar en principio estos mismos procesadores. De hecho que los utilicen no es prueba de compatibilidad si no cumplen una serie de requisitos adicionales. Los ordenadores compatibles se dicen «compatibles XT» o «compatibles AT» dependiendo si lo son con uno u otro modelo de IBM. Muchos de ellos llevan en vez del procesador 8088 el 8086 también de Intel y que tiene la ventaja de ser más rápido. Las frecuencias de reloj (en Mhz.) determinan la velocidad de procesamiento en un ordenador; por lo tanto, si un compatible se ofrece como más rápido que el IBM; hay que comprobar que efectivamente en las especificaciones técnicas lo indique con toda claridad. Pueden surgir algunos problemas al ejecutarse programas escritos específicamente para las velocidades del IBM, para solucionar este pequeño problema hay marcas que ofrecen estos modelos con un conmutador que permite pasar de la frecuencia de IBM a una mayor por el mismo usuario. Fuera de este pro-

commodore *Magazine*

La revista imprescindible para los usuarios de los ORDENADORES PERSONALES COMMODORE.

COMMODORE es una publicación mensual que le ayudará a obtener el máximo partido a su máquina.

GRATIS PARA USTED

Si se suscribe a ***COMMODORE MAGAZINE***

Una obra imprescindible en la biblioteca de todo poseedor de un ordenador personal ***COMMODORE.***

TITULO: ***LOS MEJORES PROGRAMAS PARA EL COMMODORE 64***

Un regalo de ***200 páginas,*** *tamaño 15,5 x 21,5 cuyo precio de venta es de*

800 PTAS.



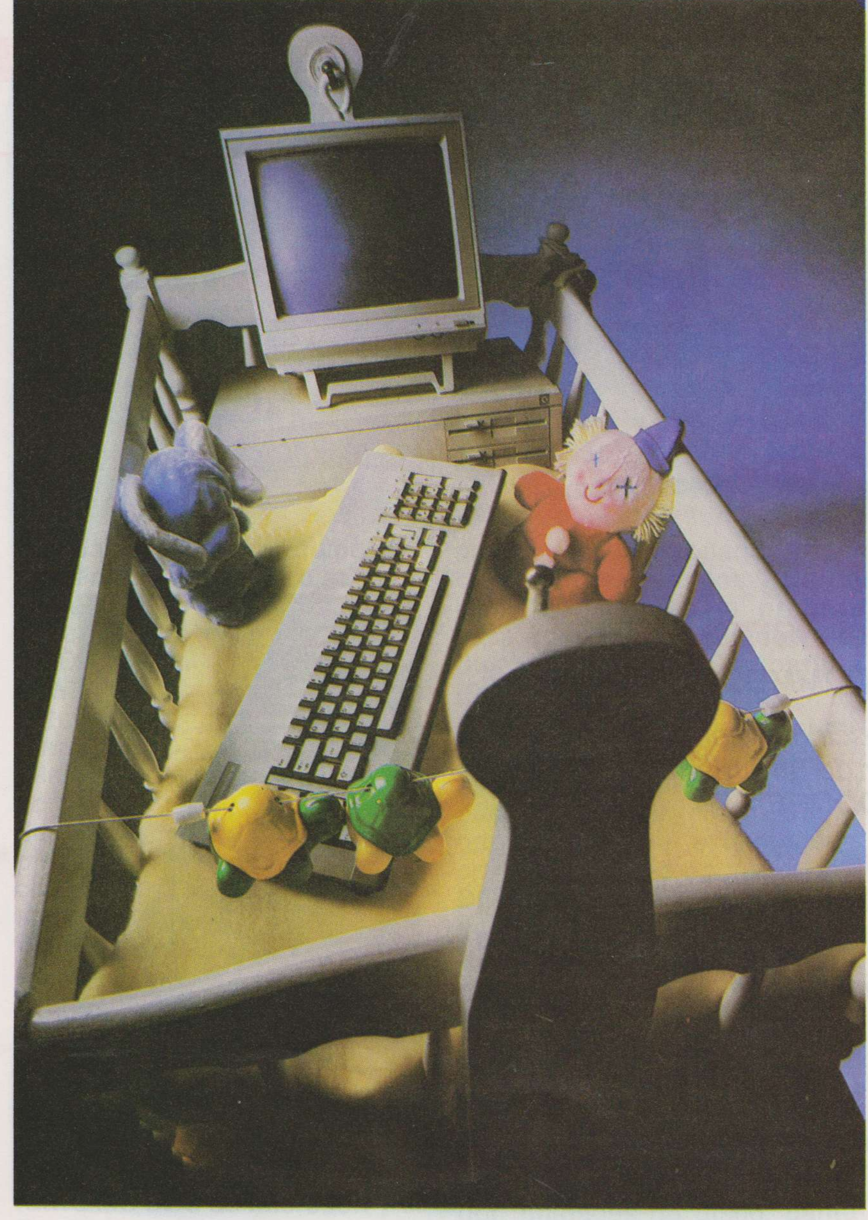
ADEMAS, le hacemos un 15% de descuento sobre el precio real de suscripción (12 números).

PRECIO REAL DE SUSCRIPCION	OFERTA ESPECIAL	AHORRO
~~3.600~~ PTAS.	**3.060** PTAS. (IVA INCLUIDO)	**540** PTAS. + LIBRO DE REGALO

APROVECHE ahora esta irrepetible oportunidad para suscribirse a COMMODORE MAGAZINE. Envíe HOY MISMO la tarjeta de suscripción adjunta a la revista que no necesita sobre ni franqueo. Deposítela en el buzón más cercano. Inmediatamente recibirá su primer ejemplar de COMMODORE MAGAZINE más el REGALO.

commodore *Magazine*

Bravo Murillo, 377
Tel.: 733 79 69
28020 MADRID



Commodore PC 10

en un compatible, éste corre perfectamente en un IBM (o viceversa) sin ningún cambio, con la ventaja adicional, como se verá más adelante, de que se puede emplear el mismo diskette en ambos equipos. Sin embargo, el BIOS (sistema básico de entrada/salida) del sistema operativo es una parte del mismo que tiene registro de propiedad. Por lo tanto, IBM tiene registrado su BIOS (que reside en memoria ROM); esto lleva al criterio de que no puede haber ninguna máquina que sea 100% compatible con IBM, por no poder copiar este BIOS sin infringir las leyes pertinentes. Sin embargo, los sistemas operativos de los compatibles cumplen las mismas funciones, y si se presentan algunos problemas puede ser a nivel de programación en ensamblador; en lo que respecta a programas de gestión, bases de datos, tratamientos de textos, etc., no se presenta ninguna compatibilidad. Hay que señalar que la mayoría de ellos se pueden arrancar perfectamente con el PC-DOS de un IBM, e incluso en este aspecto hay marcas que llegan aún más lejos y venden sus equipos sin sistema operativo indicando que se debe arrancar con uno de IBM.

blema no hay ningún otro a este nivel, los microprocesadores son fabricados por Intel y cualquiera los puede comprar para construir ordenadores. El Intel 80286 es un procesador mucho más potente que el 8088 y puede direccionar hasta 16 millones de bytes en memoria (16 Mbytes) permitiendo el desarrollo de programas de gran envergadura.

SISTEMA OPERATIVO Y BASIC

Tanto IBM como los compatibles emplean básicamente el mismo sistema operativo y el mismo Basic (ambos son de Microsoft); IBM le llama PC-DOS y las otras marcas MS-DOS (tienen licencia de Microsoft, pero no pueden usar el nombre de IBM). Las versiones actuales de IBM son el PC-DOS 3.0 y PC-DOS 3.1. Los compatibles emplean el MS-DOS 3.0, 3.1 y muchos la versión 2.11. Con el Basic ocurre algo similar, siendo la versión actual la 3.0 en la mayoría de los casos (IBM le llama BASICA y los demás generalmente emplean el GW-BASIC). Las últimas versiones incorporan mejores funciones en muchos aspectos tales como poder pasár del BASIC al DOS y regresar a él sin perder el programa de la memoria.

Los programas son compatibles; esto quiere decir que si se escribe un programa en Basic

UNIDADES DE DISKETTES

Los formatos actuales son diskettes de 360 K para XT o compatibles y de 1,2 Mb para los AT. Tanto IBM como los compatibles pueden intercambiar diskettes sin ningún problema, el formateo es el mismo en ambos casos. Un diskette puede ser formateado en un equipo y leído en otro sin que se presente ningún problema salvo los que se originen por la versión del sistema operativo (la versión 2.0 no puede leer formatos del 3.0, pero lo inverso sí es perfectamente válido). Incluso hay muchos fabricantes independientes que ofrecen unidades de diskettes para añadirlos a cualquier PC de IBM o compatibles, lo que permite gran flexi-

bilidad a la hora de comprar un ordenador. Al principio se puede comprar una configuración básica con una unidad de diskette y posteriormente comprar en las muchas casas que ofrecen periféricos para compatibles cualquier unidad de diskette adicional.

Lo mismo se aplica a los discos duros (discos de gran capacidad que no se miden en Kbytes sino en Mbytes, o sea, en millones de bytes). Los XT tienen actualmente capacidad de 20 Mbytes (modelos anteriores tenían 10 Mb), los ATs tienen discos duros de 30 Mbytes (antes 20 Mbytes). Hay compatibles que ofrecen en este aspecto discos duros de más capacidad o como en el caso de los diskettes a medida de que surjan las necesidades se pueden comprar discos duros de 40, 60 o incluso 120 Mbytes. En este aspecto hay mucha variedad en las ofertas.

MEMORIA ESTANDAR AMPLIACIONES

Hasta ahora el lector tiene información que le permite ir haciéndose una idea de las ventajas o desventajas que ofrecen los compatibles respecto a los micros de IBM. La oferta de las diferentes marcas juega fundamentalmente con ofrecer más capacidad de almacenamiento a menor precio. Continuando con el estudio de los mismos hablaremos un poco de la memoria, centrándonos fundamentalmente en el modelo XT (dejaremos el AT para próximos artículos). Los sistemas operativos PC-DOS o MS-DOS pueden direccionar hasta un máximo de 640 Kbytes, aunque hay formas de superar esta «barrera del DOS». Muchos programas necesitan en la actualidad 256 ó 384 K para poder trabajar. Hay que señalar que el Basic estándar tanto en el IBM como en los compatibles sólo requiere 64 K (hay actualmente Basics como el Quick Basic de Microsoft que aprovechan toda la memoria disponible).

La placa principal de los micros puede albergar determinada cantidad de memoria (en chips de 64 ó 256 bytes), los modelos antiguos del IBM sólo permitían 256 K como máximo, los actuales permiten hasta 640 K. Esto es un dato importante a considerar, y hay que recordarlo como memoria estándar y como memoria máxima ampliable en la placa base. Por ejemplo, si se compra un modelo con 256 K de memoria estándar ampliable hasta 640 (todos pueden ampliarse hasta las 640 K con las que trabaja el sistema operativo), hay que saber si se puede ampliar sobre la placa principal (comprando sólo los chips de memoria) o si por el contrario se necesita una tarjeta adicional (más cara que comprar sólo los chips). En resumen, es necesario saber la cantidad de memoria con la que se entrega el equipo y cuánto cuesta ampliarlo hasta 256, 384 ó 640 Kbytes. La mayoría de los programas de base de datos o de hojas de cálculo requieren en la actualidad entre 256 y 380 K. Si sólo se va a trabajar con el Basic basta con tener 128 K en total (aunque casi ningún micro se ofrece con tan poco en la actualidad). Es importante saber



Commodore PC 20

con qué programas se va a trabajar al principio para determinar la configuración mínima necesaria.

MONITORES Y GRAFICOS

No todos los programas pueden hacer gráficos, los que pueden hacerlo necesitan una tarjeta gráfica monocromática o a color. No siempre se necesita esta facilidad. Los monitores pueden ser monocromáticos (fósforo verde o ámbar) o a color (de diferentes resoluciones). El IBM XT se ofrece con pantalla verde y sin tarjeta gráfica o con monitor a color con su respectiva tarjeta gráfica. Muchos compatibles ofrecen en su configuración estándar una tarjeta gráfica y pantalla verde. De todas formas si no se tienen estas facilidades es posible comprar una tarjeta de este tipo en cualquier casa que ofrezca periféricos para IBM y compatibles.

Estas tarjetas son muy fáciles de instalar (al igual que las de ampliación de memoria). Todos los equipos (IBM y compatibles) llevan unas ranuras en la placa principal (se les conoce como «slots») que permiten añadir tarjetas de memoria, gráficas, multifunción (memoria, reloj, salidas serie, etc.), comunicaciones, etc., sin ninguna complicación. Generalmente el color es un lujo que puede elevar el precio del equipo desde unas 40.000 a 80.000 pesetas.

SOFTWARE COMPATIBLE

La mayoria del software disponible para los IBM corre sin problemas en cualquier compatible. Sin embargo, muchos programas especifican que corren en IBM PC/XT/AT o 100% compatibles. En estos casos es necesario pedir confirmación en la casa que ofrece un compatible o llamar a la casa distribuidora de los programas para preguntarle concretamente la fiabilidad de su producto en una determinada marca y modelo de ordenador. Los programas principales que se emplean en un PC son los de gestión (contabilidad, nómina, almacen, facturación, clientes y proveedores), bases de datos, hojas de cálculo, tratamiento de textos y los lenguajes de programación (Basic, Cobol, Fortran, Ensamblador, C, Pascal, etc.). De todos ellos hay multitud de versiones y de categorías. Los precios oscilan entre unos cuantos miles de pesetas hasta más de 600.000 pesetas. Dejaremos para otro artículo la definición y catalogación de los mismos y de las múltiples prestaciones que ofrecen y que muchos de ellos justifican por sí solos la compra de un ordenador. Para terminar esta parte baste con

decir de que es necesario definir desde el principio (cosa difícil) qué tipo de trabajos se van a realizar con el ordenador para determinar el coste de inversión en programas y tener siempre presente de que el software tiene un precio y debe considerarse como parte del gasto que va a realizarse. El lector no debe olvidar que gran parte del mérito de tener un ordenador personal IBM o compatible radica en la calidad y potencia de los programas que corren en ellos. El mercado demanda en la actualidad más y mejor software estándar o a medida que sólo alcanza la calidad exigida en equipos con capacidad de memoria y de almacenamiento en disco como los que ofrecen los PCs. De ahí la importancia en la selección de los programas o de los lenguajes de programación que se empleen.

El lector debe considerar seriamente lo que necesita en la actualidad y lo que puede necesitar en el futuro. Puede empezar con una configuración básica (un diskette y 256 K de memoria) y poco a poco ir comprando las ampliaciones que necesite (otra unidad de diskette, un disco duro, etc.). No sólo debe fijarse en los precios del modelo que se le oferta, sino que debe además pedir que le digan exactamente cuánto le costaría ampliarlo a una configuración más grande. Esto le permitirá comprar sólo lo que necesita y no pagar por prestaciones que no va a emplear (monitor a color, gráficos, etc.). La impresora generalmente se compra aparte y hay más de 20 marcas en el mercado perfectamente compatibles y de todo precio (desde 28.000 hasta 500.000 pesetas.).

Prácticamente todas las casas de ordenadores ofrecen compatibles y para poner algunos ejemplos citamos a marcas como Commodore, Amstrad, Spectravideo, Sperry, HP, Olivetti, Televideo, Copam, Victor, Unitron, Toshiba, Kaypro, Compaq, Xerox, Tulip, Wang, Mark, Multitech, NCR, Pro-Fax, Zenith, Bondewell, Cannon, Ericcson, ITT, Keystone, Mark, etc. Algunos son americanos, a otros se les conoce como los taiwaneses por su procedencia de Taiwan; unos pocos son de Corea, Holanda, Alemania, etc. Los precios se encuentran en la gama que va desde las 180.000 pesetas hasta casi el millón. La guerra de precios que mantienen todas las casas ha hecho bajar constantemente los precios y todos los pronósticos apuntan a un descenso aún mayor, lo que llevaría rápidamente a la venta masiva de este tipo de ordenadores personales.

Particularmente hablando de



Amiga, un microordenador que puede ser compatible.

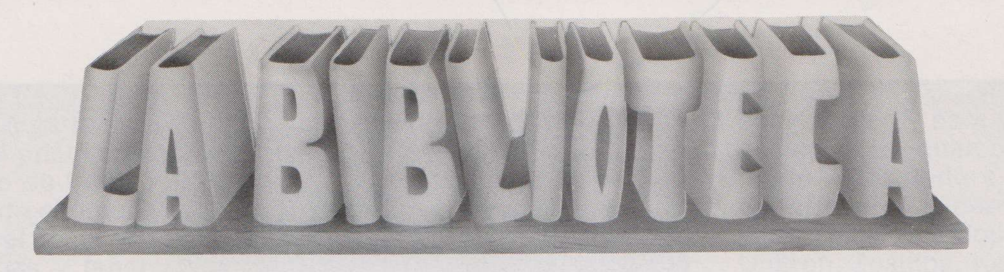
los modelos de Commodore señalaremos que en la actualidad se ofrecen dos modelos: el Commodore PC 10 II y el Commodore PC 20 II. Ambos se ofrecen con una configuración básica de 512 K de memoria central, tarjeta gráfica de alta resolución, monitor monocromático, salida serie y paralela, 5 slots libres para ampliaciones (tarjetas de comunicaciones, tarjetas con varias salidas serie o paralela, memoria, etc.). El primer modelo tiene dos unidades de diskettes y su precio está sobre las 300.000 pesetas. El segundo ofrece un disco duro de 20 Mbytes y está sobre las 500.000 pesetas. El sistema operativo es el MS-DOS, última versión que incorpora utilidades de spool de impresora y de SORT para ficheros y un Basic avanzado con capacidad de gráficos, sonido y ficheros de acceso directo. El teclado es compatible con el de IBM, aunque mejora la disposición de las teclas y lleva de forma estándar indicadores luminosos de caps locks y num locks.

A nivel de software es compatible con programas tan populares como el dBASE III y Framework de Asthon-Tate, 1-2-3 y Symphony de Lotus, WordStar, compiladores de Fortran, C, Pascal, Basic, Cobol, Lisp, etc. Corre asimismo prácticamente todos los programas de Gestión que se ofrecen en la actualidad.

CONCLUSIONES

Esperamos que nuestros lectores tengan después de leer este artículo una idea más clara del fenómeno de los compatibles. La gran difusión de este tipo de ordenadores de 16 bits no quiere decir que todos los micros que no cuenten con estas características sean ya obsoletos. Hay que aplicar un criterio práctico en este sentido y entender que un ordenador por grande o pequeño que sea nos puede prestar valiosa ayuda en nuestro quehacer diario siempre y cuando sepamos sacar el máximo provecho del mismo. Hay que obtener una rentabilidad de esta herramienta aprovechándola al máximo. Esto se logra con imaginación y tratando en mayor o menor medida de usar los diferentes programas a nuestro alcance para que faciliten nuestras tareas diarias. En artículos sucesivos hablaremos de lenguajes de programación, software para compatibles y muchos otros temas que son también de interés para cualquiera que tenga un ordenador personal en la casa o en oficina.

Nos queda sólo recomendar a nuestros lectores que sean críticos a la hora de evaluar un nuevo equipo y que no sólo se fijen en los precios de la configuración que se les oferte, si no que además prevean los gastos futuros de ampliación del mismo y traten de determinar con la mayor precisión posible las necesidades reales actuales y futuras.



LOGO, de la tortuga a la inteligencia artificial
Autor: Luis Rodríguez-Roselló.
Editorial: Vector Ediciones.
Páginas: 581

El interés que el lenguaje LOGO despierta en todo el mundo es indudable. Existen publicaciones periódicas dedicadas en exclusiva a este lenguaje, congresos dedicados íntegramente al LOGO; se crean asociaciones de usuarios en muchos países. Y sin embargo, y a pesar de que todo el mundo hable del LOGO, sigue siendo un gran desconocido, y la idea más generalizada que sobre el mismo se tiene es que se trata de un lenguaje para niños.

Es por tanto, con la actual incorporación de la informática a la enseñanza, cuando se produce un momento idóneo para iniciar una reflexión que permita conocer a fondo este lenguaje, tanto desde un punto de vista informático como desde su filosofía educativa y sus posibilidades reales en la enseñanza. Este es el objetivo primordial que se propone el presente libro.

El enfoque dado por el autor proporciona una visión general del LOGO, haciendo un especial hincapié en las estructuras de los programas, alejándose de un enfoque de tipo «manual» para un ordenador concreto.

El planteamiento, por otra parte muy acertado, del libro, supone que el lector ignora todo sobre este lenguaje, por lo cual comienza tratando los aspectos generales del LOGO, su pedagogía asociada y su relación con el mundo de la informática en la educación, para a continuación pasar al aprendizaje concreto del lenguaje. Presenta los procedimientos incorporados, las entradas que necesitan estos procedimientos, la posibilidad de que el usuario defina sus propios procedimientos, las variables, las listas, el



control del editor y la sintaxis.

El siguiente capítulo nos introduce en la parte del LOGO más conocida, y quizá la más espectacular: los gráficos de tortuga. Aprendemos las instrucciones que permiten subir y bajar la pluma, mostrar y esconder la tortuga, situarla en coordenadas cartesianas, orientarla; igualmente podemos pedirle al ordenador que nos proporcione información sobre su situación en un momento dado, el color de la pluma y otros datos.

Los gráficos de tortuga nos servirán también para comprender la mejor cualidad del lenguaje LOGO: la programación modular. Una serie de ejemplos nos muestran cómo diseñar módulos que realizan cada uno un dibujo, y luego combinarlos de forma que se interrelacionen. De esta forma podremos definir una serie de procedimientos para dibujar una casa, un árbol, un coche, una estrella, y con ellos dibujar un paisaje de ciudad.

Luego podremos enfrentarnos a cosas más abstractas, como son las operaciones lógicas, las funciones matemáticas incorporadas, la definición de funciones o la representación de números y operaciones aritméticas.

Sin embargo, y como se puede comprobar, el LOGO no es solamente los gráficos de tortuga, además es un lenguaje bastante potente en lo que respecta al tratamiento de palabras como grupos de letras, y de listas como grupos de palabras. El autor nos demuestra cómo extraer un elemento cualquiera de una lista, cómo ampliarla, cómo calcular el número de elementos. También se pueden realizar funciones más complejas, como insertar una lista en otra, reemplazar un elemento de una lista por otro, realizar operaciones matemáticas con listas de números e incluso operaciones con números romanos.